Advancing Large-Scale Molecular Dynamics through Machine Learning

Théo Jaffrelot Inizan

Sorbonne Université

École Doctorale de Chimie Physique et de Chimie Analytique de Paris Centre
Laboratoire de Chimie Théorique - UMR 7616 CNRS

Advancing Large-Scale Molecular Dynamics through Machine Learning

Théo Jaffrelot Inizan

Thèse de doctorat de Chimie Théorique

Présentée et soutenue publiquement le 03/10/2023

devant un jury composé de:

Reviewers	Carine Clavaguera	Université Paris Saclay
	Markus Meuwly	Universität Basel
Examiners	Alessandra Carbone	Sorbonne Université
	Jérome Hénin	Institut de Biologie Physico-Chimique
	Nathalie Lagarde	Conservatoire national des arts et métiers
	Louis Lagardère	Sorbonne Université
	Gabriel Stoltz	École des Ponts ParisTech
Supervisor	Jean-Philip Piquemal	Sorbonne Université

Résumé

Introduction

La compréhension du comportement des molécules au niveau atomique est au coeur de la physique et de la chimie. Le progrès des architectures de calcul, des algorithmes et du calcul haute performance a grandement contribué au développement des méthodes de simulation. Dans les années 1960, les travaux de Karplus, Verlet et Rahman ont donnés naissance à la dynamique moléculaire.[1, 2, 3] La dynamique moléculaire a grandement contribué à l'essor des simulations pour les systèmes chimiques, biochimiques mais également pour les matériaux. Cependant, cette approche est confrontée à plusieurs limitations.

Tout d'abord, pour les systèmes de grande taille, i.e. comme ceux de la biologie, il est impossible d'utiliser des méthodes de type structure électronique, car trop coûteuses en temps de calcul. Cela a conduit à la création de modèles empiriques basés sur des formules physiques simplifiées et paramétrisées sur des bases de données de valeurs de référence, appelés champs de forces. D'un côté, les champs de forces dits "classiques" ont un faible coût de calcul mais manquent de précision. De l'autre, les champs de forces dits polarisables permettent de prendre en compte des effets physiques supplémentaires comme la polarisation et les effets à plusieurs corps mais au détriment d'un coût de calcul plus élevé. Ainsi, malgré le fait que ces modèles aient ouvert la voie à l'étude par dynamique moléculaire de systèmes toujours plus complexes[4, 5], ils reposent toujours sur des modèles physiques simplifiés ne permettant pas de prendre en compte de façon précise la mécanique quantique.
Parallélement, une autre limitation de la dynamique moléculaire est reliée aux échelles de temps accessibles. En effet, les temps accessibles en dynamique moléculaire sont de l'ordre de la microseconde (milliseconde dans de rares cas) alors que les temps caractéristiques des processus chimiques, par exemple biologiques, peuvent être de l'ordre de la seconde. Cette problématique a engendré le développement d'algorithmes dits d'échantillonnage accéléré qui ont pour but d'aider les simulations à franchir les barrières énergétiques, accélérant ainsi l'exploration de l'espace conformationnel et permettant d'observer des évènements rares.
De plus, la haute-dimensionnalité des trajectoires résultantes des simulations pose des défis pour l'analyse. En effet, l'extraction de certaines propriétés, directement comparables aux données expérimentales, telles que les taux de transition entre macro-états et les profils d'énergie libre, est un champs de recherche à part entière, un exemple étant les modèles d'états de Markov.

L'objectif de cette thèse vise à résoudre certaines de ces limitations par l'utilisation et le développement de modèles d'apprentissage profond. En couplant l'apprentissage profond aux méthodes de dynamique moléculaire il est possible d'améliorer la précision des champs de forces, accélérer les simulations et comprendre de façon plus fine les processus chimiques. Ces dernières années, la puissance grandissante des machines parallèles, des architectures de type cartes graphiques (GPU) ainsi que la disponibilité à de vastes ensembles de données ont ouvert la voie à de nouveaux paradigmes dans l'apprentissage profond. L'entraînement de larges modèles d'apprentissage profond entrainés sur des millions, voire millards, de données est devenu courant. En particulier, dans le cadre de la dynamique moléculaire, d'importants efforts ont été consacrés au développement de modèles de type champs de forces basés sur l'apprentissage profond, nommés par la suite potentiels neuronaux. L'idée est d'utiliser un modèle d'apprentissage profond, souvent composé d'unités appelées neurones, afin d'obtenir une relation mathématique directe entre les positions atomiques et l'énergie potentielle. L'entrainement de ce type de modèle utilise des données composées de petites molécules dont l'énergie potentielle a été calculée par des méthodes de structure électronique. L'engouement pour ce type de modèles vient de la capacité du réseau de neurones à se généraliser à des systèmes chimiquement très différents et de plus grande taille. De plus, son coût de calcul raisonable le place comme un bon candidat pour résoudre l'un des graal de la chimie quantique: simuler avec précision quantique des systèmes de plusieurs millions d'atomes.

Cependant, ces potentiels neuronaux restent basés sur une forme fonctionnelle locale et ont tendance à négliger les interactions à longue portée. Ces interactions sont cruciales pour de nombreux systèmes comme la phase condensée (i.e l'eau liquide par exemple), les protéines ou l'ADN. De plus, ils ont des difficultés à traiter de façon précise les molécules chargées, les solvants et ne prennent pas en compte, a priori, les effets quantiques des noyaux. De surcroît, l'intégration d'une plateforme multi-GPU compatible avec les modèles d'apprentissage profond dans les codes de dynamique moléculaire n'est pas évident et représente un autre défi à relever. Alors que de nouveaux modèles d'apprentissage profond sont développés chaque jours par une grande communauté de développeurs et d'utilisateurs via Python et ses bibliothèques dédiées, telles que PyTorch, TensorFlow, Scikit-learn et Keras,[6, 7, 8, 9], la plupart des codes de dynamique moléculaire comme CHARMM, GROMACS et Tinker-HP [10, 11, 12, 13] utilisent des langages compilés tels que Fortran ou C++. Par conséquent, combiner les modèles d'apprentissage profond et les codes de dynamique moléculaire de façon efficace n'est pas une tâche facile.

Pour aborder ces problèmes, le Chapitre 1 fait un état de l'art dans le domaine de la chimie computationnelle. La Section 1.1 explique les limitations de l'équation de Schrödinger et introduit un grand nombre de méthodes de chimie quantique, la plupart de ces méthodes étant utilisées pour générer les jeux de données ou servant de référence pour analyser la précision de certains modèles. La Section 1.2 analyse en profondeur les modèles de type champs de forces. Une attention particulière a été donnée pour discuter leurs limitations afin de pouvoir mettre en oeuvre une stratégie pour les améliorer. Un aperçu des différents modèles d'apprentissage profond ainsi que leurs utilisations au sein des champs de forces est présenté en Section 1.3. Enfin, la Section 1.5 donne une vue d'ensemble des différentes techniques d'échantillonnage accélérées et des récents couplages avec l'apprentissage profond.

Pour aborder les problèmes de la dynamique moléculaire énoncés précédement, la Section 2.2 du Chapitre 2, commence par introduire Deep-HP, une plateforme multi-GPU au sein du logiciel Tinker-HP, Appendix 2.1, qui permet la construction de champs de forces hybridés avec l'apprentissage profond. Cette plateforme évite les transferts de données hôte-périphérique, détrimentaux à la vitesse de calcul, en convertissant les pointeurs des composantes tensorielles des modèles d'apprentissage en pointeurs GPU Fortran via une interface Python/C++. Les capacités de la plateforme sont testées par la simulation à grande échelle des protéines du SARS-CoV-2 comme la protéase principale ou M^{pro} et la protéine Spike avec l'utilisation de potentiels neuronaux.
S'appuyant sur la plateforme Deep-HP, le Chapitre 2, Section 2.1, présente le modèle Deep Neuronal Network Many-Body Dispersion (DNN-MBD) model, un modèle de dispersion stochastique à plusieurs corps à complexité linéaire.[14] Dans ce modèle, l'apprentissage profond est utilisé de sorte à contourner des calculs coûteux de densité électronique. Ce modèle permet de prendre en compte de façon très précise les interactions de dispersion, tout en pouvant être utilisé pour simuler des systèmes de plusieurs millions d'atomes. Ceci permet d'ouvrir la voie à la compréhension des effets d'interactions à plusieurs corps sur les systèmes biochimiques. En plus d'éviter les calculs de densité électronique, le modèle d'apprentissage profond permet également d'améliorer la précision intrinsèque de l'approche MBD et d'augmenter sa transférabilité. En effet, bien que le modèle ait été entrainé sur des molécules organiques, il peut être utilisé dans les domaines de la biochimie et de la science des matériaux. Son caractère universel lui permet également d'être combiné avec de nombreux modèles physiques comme les méthodes de structure électronique, les champs de force ou encore les modèles d'apprentissage profond.
Une autre approche explorée dans cette thèse consiste à utiliser l'apprentissage profond pour améliorer la précision des modèles de champs de forces polarisables existants. La Section 2.2, présente un modèle hybride qui couple le potentiel neuronal ANI-2X avec le champs de forces polarisable AMOEBA. Le premier, ANI-2X est utilisé pour les interactions protéine-protéine alors que le dernier est utilisé pour modéliser le reste des interactions. Ce modèle hybride a l'avantage de conserver les interactions à longue portée d'AMOEBA qui sont très précises, et celles à courte portée d'ANI-2X. Dans cette section nous mettons également en lumière le développement de techniques d'intégration multi pas-de-temps permettant d'accélèrer grandement les simulations. Le couplage entre ANI-2X, AMOEBA et les intégrateurs multi pas-de-temps a été validé par le calcul d'énergie de solvatation de 70 molécules dans divers solvants ainsi que l'énergie libre d'interaction hôte-invité de 14 systèmes hôte-invité issus des compétitions du SAMPL.
La Section 2.4 présente le modèle Q-AMOEBA-NN, un modèle d'apprentissage profond basé également sur AMOEBA. Dans un premier temps, le modèle combine les interactions à longue portée d'AMOEBA, avec l'adaptive Quantum Thermal Bath (adQTB) permettant d'approximer de façon précise les effets quantiques des noyaux (NQE) tout en ayant un coût de calcul supplémentaire négligeable. Ceci a été rendu possible grâce au développement de la plateforme Quantum-HP, brièvement introduite en Section 2.3. Les interactions à courte portée sont prises en compte par un potentiel neuronal basé sur un nouveau type d'architecture de réseaux neuronaux équivariants. Le développement du modèle Q-AMOEBA-NN a été rendu possible grâce à la paramétrisation AMOEBA d'une vaste base de données contenant des millions de conformations, comprenant des

dipeptides, des dimères, des "clusters" d'eau et des ions solvatés. Les performances de ce modèle sont évaluées sur diverses propriétés, telles que des balayages d'énergie potentielle et des profils d'énergie libre.
Une part importante de cette thèse a également été consacrée au développement d'algorithmes d'échantillonnage accélérés, visant à renforcer l'exploration de l'espace des conformations des systèmes biochimique. L'un des algorithmes, décrit dans le Chapitre 3, Section 3.1, est la technique d'échantillonnage adaptatif (AS). La technique AS fonctionne par itérations de simulations indépendantes de dynamique moléculaire lancées en parallèle. Le processus de sélection de l'algorithme AS repose sur une projection en basse dimension des coordonnées cartésiennes du système. Cette projection peut être obtenue grâce à divers modèles d'apprentissage profond de réduction de dimension, tels que les autoencodeurs ou l'analyse en composantes principales. L'efficacité de l'algorithme AS a été démontrée en échantillonnant l'espace des conformations de la protéine M^{pro} du SARS-CoV-2, permettant de générer plus de 50 μs de simulations. A ce jour, c'est la plus longue simulation réalisée avec un champ de forces polarisable.
Plusieurs couplages entre l'algorithme AS et diverses nouvelles méthodes de dynamique moléculaire accélérée par processus gaussien, spécifiquement conçues pour les champs de forces polarisables et les intégrateurs multi pas-de-temps, sont présentées dans le Chapitre 3, Section 3.3. Il est montré que ces nouveaux couplages permettent d'accélérer la vitesse de calcul d'un ordre de magnitude. La validation de ces méthodes a été faite en calculant des profils d'énergie libre de plusieurs protéines, par exemple le CD2-CD58, démontrant ainsi son efficacité dans l'échantillonnage de systèmes complexes.
Nous avons donc exploré comment l'apprentissage profond peut améliorer la précision des champs de forces et accélérer l'exploration de l'espace des conformations. Un autre aspect, tout aussi important, abordé dans la Section 3.2 du Chapitre 3 et en Appendix B est l'analyse des données de simulation à l'aide de techniques d'apprentissage profond. Divers algorithmes de partitionnement de données, "clustering", et de modèles d'états de Markov basés sur des méthodes d'apprentissage profond ont été utilisés afin d'extraire des caractéristiques structurelles de systèmes biologiques. Grâce à ces analyses, des informations précieuses sur les mécanismes de liaison entre médicaments et protéines ainsi que des interactions allostériques, généralement difficiles à capturer, ont pu être mieux comprises.

Les recherches menées lors de cette thèse illustrent les différentes applications de l'apprentissage profond en dynamique moléculaire, englobant des efforts multiples visant à améliorer la précision des champs de forces, les techniques d'échantillonnage et à aider à analyser les résultats de simulations. L'objectif principal est d'ouvrir de nouvelles voies en dynamique moléculaire via l'exploitation des modèles d'apprentissage profond, de la mécanique statistique, de la mécanique quantique et du calcul parallèle sur GPU. Grâce au futur modèle Q-AMOEBA-NN, ainsi qu'aux résultats obtenus au cours de cette thèse, nous espérons que des simulations avec précision quantique contenant des millions d'atomes seront possibles dans un avenir proche.

Conclusion

Tout au long de cette thèse, des modèles d'apprentissage profond ont été utilisés à différents niveaux afin de tenter de résoudre certaines limitations de la dynamique moléculaire. Alors que les champs de forces offrent un faible coût de calcul, ils ne peuvent, néanmoins, capturer un certain nombre d'effets quantiques. De surcroît, l'accélération d'échantillonnage reste un défi en-soi et encore plus particulièrement pour les systèmes biochimiques. Cette difficulté est intrinséquement liée aux échelles de temps caractéristiques des phénomènes présents dans systèmes biochimiques qui dépassent largement l'échelle de temps accessible par dynamique moléculaire.

Ces dernières années, d'importants efforts ont été faits pour incorporer l'apprentissage profond dans les champs de forces, l'apprentissage profond offrant la promesse de combiner la précision des méthodes de structure électronique et la vitesse de calcul des champs de forces. Alors que les méthodes de structure électronique, coûteuses mais basées sur des principes physiques rigoureux, les rendant très précises et facilement transférables, les champs de forces reposent quant à eux sur des formules physiques simplifiées les rendant peu chers en temps de calcul avec à la clé une précision et une tranférabilité dégradées. Parallélement, la plupart des potentiels neuronaux ont tendance à négliger les effets à longue portée, essentiels pour simuler avec précision la phase condensée et la structure des protéines. De plus, ils négligent pour la plupart les effets quantique des noyaux. Compte tenu de ces considérations, dans cette thèse, plusieurs modèles basés sur l'apprentissage profond et tenant compte de modèles physiques pré-éxistant ont été introduits.

Le premier modèle développé, DNN-MBD,[15] utilise un modèle d'apprentissage profond pour contourner les calculs de densité électronique. En particulier, il est entraîné sur les volumes atomiques présent dans un jeu de données, ANI-1X, composé de petites molécules organiques comportant entre 1 à 8 atomes lourds. La précision et la transférabilité du modèle DNN-MBD sont évalués sur des jeux de données de référence: S66x8 et S22. Le couplage du modèle DNN-MBD avec des fonctionnelles DFT courantes, telles PBE et PBE0, montre des précisions comparables à des méthodes de chimie quantique de référence (CCSD(T)/CBS). Combinée avec PBE0, l'erreur absolue en moyenne (MAE) sur les jeux de données S66x8 et S22 sont respectivement de 0.25 kcal/mol et 0.41 kcal/mol. Ces erreurs sont parmis les plus faibles pouvant être trouvées dans la littérature. Ces avancées ouvrent ainsi la voie à la génération de jeux de données étendus et hautement précis par l'utilisation de ce modèle. Cela offre également de nouvelles perspectives dans le domaine du développement de méthode de dispersion et quand à l'étude à grande échelle de ces effets, par exemple dans les protéines.

Une grande attention a été consacrée afin de combiner les potentiels par apprentissage profond avec les champs de forces polarisables. Pour cela, il a été nécessaire de développer une plateforme

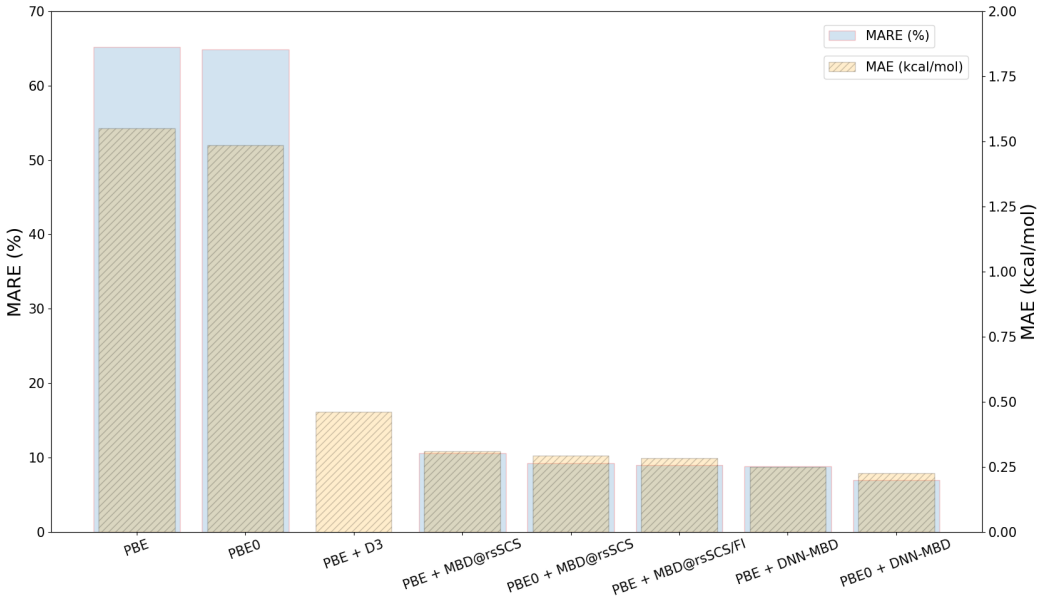


Fig. 1.: L'erreur relative absolue en moyenne (MARE) (%) et l'erreur absolue en moyenne (MAE) (kcal/-mol) sur les énergies d'interactions du jeu de données S66x8 de différents modèles de dispersion: PBE+D3[16], MBD@rsSCS[17], MBD@rsSCS/FI[18] et le modèle DNN-MBD introduit dans cette thèse.

d'apprentissage profond multi-GPU hautement parallèle et de l'intégrer dans un logiciel de dynamique moléculaire existant. La plateforme, dénommée Deep-HP, fait partie du code Tinker-HP. Cette plateforme permet aux utilisateurs de combiner des potentiels par apprentissage automatique avec les champs de forces présents dans Tinker-HP. Elle permet de faire des simulations à grande échelle avec des potentiels neuronaux. En effet, les capacités de la plateforme ont été démontrées via la simulation de grands systèmes biologique, notamment les protéines M^{pro} et Spike du SARS-CoV-2, et en utilisant un potentiel neuronal de pointe, ANI-2X, sur plusieurs centaines de GPU. Cette plateforme a conduit au développement d'un modèle hybride qui combine le potentiel neuronal ANI-2X et le champs de forces AMOEBA. Pour accélérer la vitesse de calculs du modèle hybride, des stratégies sophistiquées basées sur des intégrateurs à multi pas-de-temps ont été développées. Ces intégrateurs ont permis d'accélérer par un facteur 20 les simulations. Cela a permis d'atteindre des temps de simulations de l'ordre de la μs-ms et donc de pouvoir d'approcher les temps caractéristiques des phénomènes biologique. Pour vérifier la précision et la transférabilité du modèle pour différents environnements moléculaires, les énergies libres de solvatation de 70 molécules dans divers solvants ainsi que les énergies libres de liaison de 14 systèmes hôte-invité des compétitions du SAMPL ont été calculées et comparées aux données expérimentales. Le modèle hybride ANI-2X/AMOEBA améliore les résultats AMOEBA sur les systèmes de la compétition du SAMPL, atteignant ainsi la précision chimique ($\leq$ 1 kcal/mol). L'erreur du modèle hybride atteint 0,94 kcal/mol (vs AMOEBA: 1,81 kcal/mol).

La Section 2.4 introduit brievement le modèle Q-AMOEBA-NN. Ce modèle utilise les modèles d'interaction à longue portée du champ de forces AMOEBA et utilise un modèle d'apprentissage

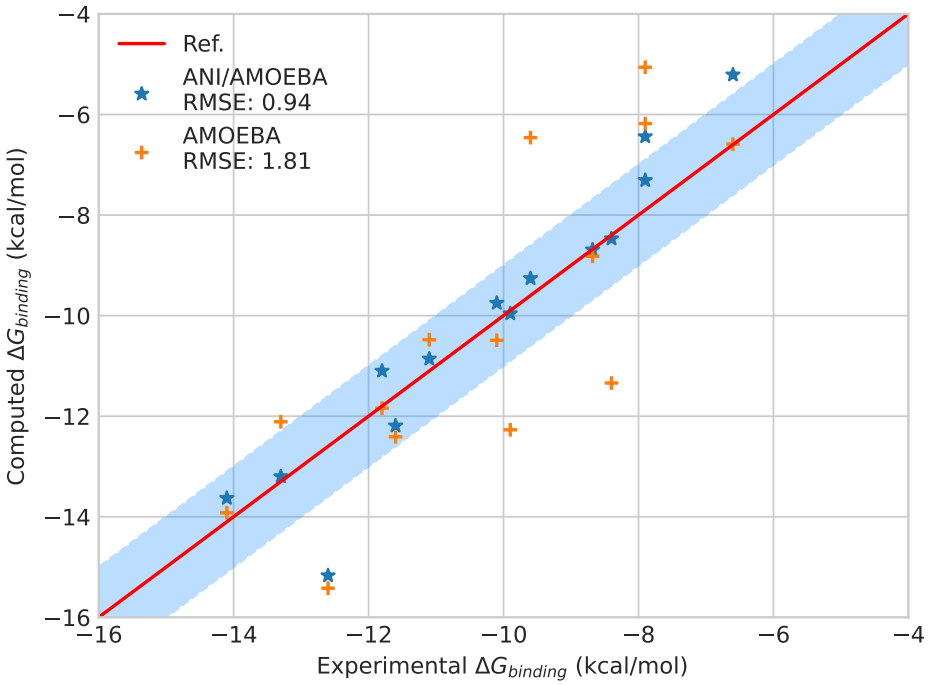


Fig. 2.: L'erreur stanard moyenne (RMSE) (kcal/mol)) sur les énergies libres des systèmes hôte-invité du "SAMPL challenge" pour le modèle AMOEBA (orange), modèle hybridé ANI-2X/AMOEBA (bleu) par rapport aux données expérimentales (rouge). Le domaine en bleu correspond à l'erreur chimique cible de 1 kcal/mol par rapport à l'expérience.

profond pour les interactions à courte portée. Afin de maintenir la stabilité et d'éviter la réactivité, les interactions d'étirement de liaison sont également décrites par AMOEBA. De plus, un modèle d'apprentissage profond supplémentaire a été utilisé pour raffiner les paramètres des interactions de van der Waals afin d'éliminer les effets quantiques des noyaux usuellement pris implicitement en compte dans les modèles existants. Le développement du modèle Q-AMOEBA-NN a été rendu possible grâce à la paramétrisation avec AMOEBA d'une vaste base de données contenant des millions de conformations, y compris des dipeptides, des dimères, des molécules d'eau et des ions solvatés. Des tests approfondis ont été menés pour évaluer les performances du modèle Q-AMOEBA-NN sur diverses propriétés thermodynamique comme l'énergie libre. En intégrant la précision quantique avec l'apprentissage profond, le modèle Q-AMOEBA-NN surmonte certaines limitations des champs de forces traditionnels mais également des potentiels neuronaux.

En plus d'améliorer la précision des champs de forces, il est tout aussi crucial d'échantillonner de façon efficace l'espace des conformations. Une partie de cette thèse se concentre justement sur le développement de techniques d'échantillonnage accélérées non-supervisées. L'une d'entre elles, appelée échantillonnage adaptatif (AS), est constituée d'itérations, où chaque itération consiste en plusieurs simulations de dynamique moléculaire lancées en parallèle. La procédure de sélection des structures de départ est une fonction de la densité des structures projetées dans un espace de

dimension réduite. Cet espace de faible dimension peut être obtenu à l'aide de divers algorithmes de réduction de dimension présents en apprentissage profond, tels que les autoencodeurs variationnels et l'analyse en composante principale. L'efficacité du modèle a été testée pour échantillonner l'espace de conformation de la protéine M^{pro} du SARS-CoV-2. Il a permis de générer plus de 50 μs de simulations de dynamique moléculaire en utilisant le champ de forces polarisable AMOEBA.
La méthode AS a permis de révéler que la protéine M^{pro} subit des changements conformationnels dûs à une interaction coopérative à longue portée entre l'interface de dimérisation et la cavité réactive, i.e un phénomène d'allostérie. Les effets de polarisation présents dans AMOEBA se sont révélés être cruciaux pour capturer ces effets à longue portée. De plus, il a été découvert que les molécules d'eau jouent un rôle structurant pour l'interface de dimérisation qui découlent des interactions de polarisation entre eau et protéine.

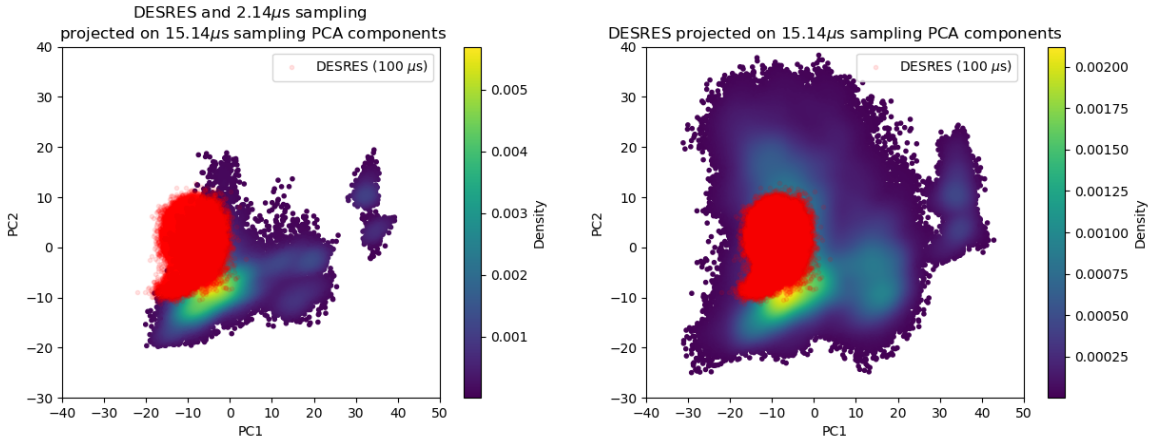


Fig. 3.: Projection des données DESRES (100μs de simulations) sur les composantes PCA de nos simulations sur la M^{pro} du SARS-CoV-2 effectuées avec le champs de force AMOEBA et la nouvelle méthode d'échantillonnage adaptatif, introduite pendant la thèse, après respectivement 2μs (gauche) et 15.14μs (droite) de simulations.

Pour améliorer davantage l'efficacité de l'échantillonnage, l'algorithme AS a été combiné avec une nouvelle méthode de dynamique moléculaire accélérée par gaussiennes (GaMD) spécifiquement conçue pour les intégrateurs à multi pas-de-temps et les champs de forces polarisables ainsi qu'avec la technique d'échantillonnage en parapluie. La combinaison de ces méthodologies d'échantillonnage permet une réduction significative du temps de calcul requis pour évaluer les profils d'énergie libre. Notamment, le couplage des trois méthodes permet d'atteindre un facteur de vitesse de 15 pour la convergence du profil d'énergie libre.

Pour conclure, cette thèse a permis de mettre en lumière plusieurs applications de l'apprentissage profond en dynamique moléculaire. Un travail important a été réalisé afin de combiner l'apprentissage profond avec des modèles physiques. Un des messages de cette thèse est que l'apprentissage profond ne doit pas remplacer totalement les modèles physiques actuels, mais plutôt être intégré de manière réfléchie à ces derniers. Par exemple, l'intégration de modèles d'apprentissage profond dans les champs de forces doit être couplé avec les modèles existants d'interactions à longue portée et d'effets

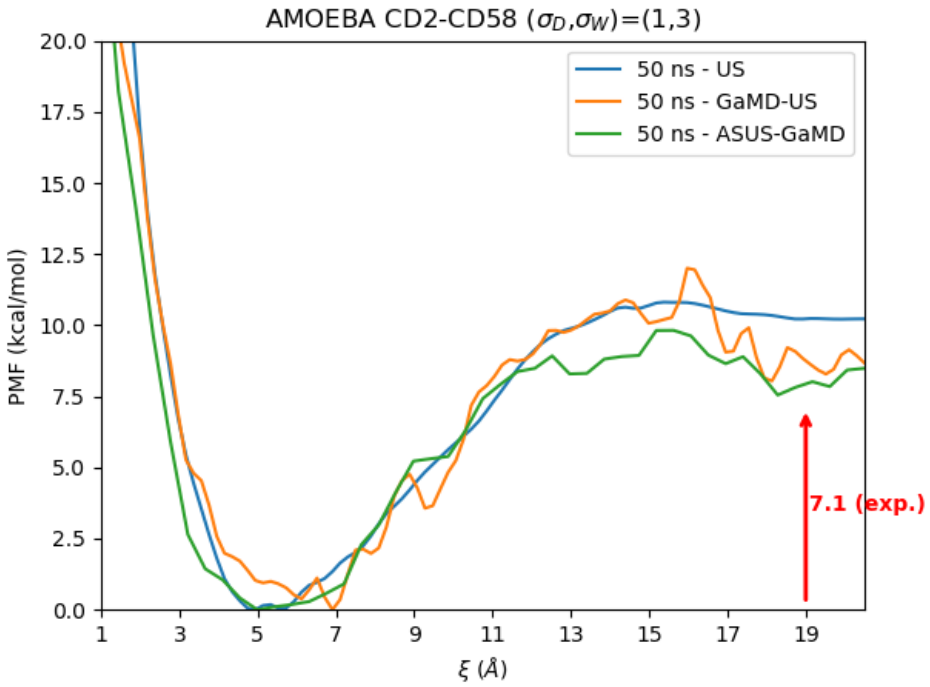


Fig. 4.: Profile d'énergie libre de dissociation du complexe CD2-CD58 obtenue avec le champs de force AMOEBA et différentes méthodes d'accélération d'échantillonnage: Umbrella Sampling (US) et deux méthodes introduites dans la thèse appelées GaMD–US et ASUS–GaMD. La flèche en rouge correspond à la valeur expérimentale.

quantiques nucléaires. Un autre exemple est le couplage avec des techniques d'échantillonnage accéléré qui doit être réalisé de sorte à ne pas oublier les étapes de débiaisage afin de pouvoir extraire des propriétés quantitatives.

Ces travaux de thèse, en plus de s'inscrire dans un contexte d'exploration de l'utilisation de l'apprentissage profond en dynamique moléculaire à grande échelle, ont également permis de mettre en place des modèles robustes désormais utilisables pour l'étude de nombreux systèmes biochimiques. Ces modèles ont déjà été adoptés par la communauté. Nous espérons que tout ceci ouvrira la voie à de nouvelles recherches sur l'utilisation de tels modèles dans des domaines d'application variés.

Abstract

Molecular dynamics simulations provide valuable insights into the behavior of molecular systems at the atomic level. However, large-scale Molecular Dynamics simulations face some limitations. Firstly, the use of physically-motivated empirical force fields may not accurately capture some quantum mechanical effects (reactivity, quantum nuclear effects etc...). Secondly, the sampled timescale in simulations is often shorter than the timescale of the processes of interest, requiring the use of enhanced sampling algorithms to overcome energy barriers. Lastly, the high-dimensional nature of simulation trajectories can pose challenges for interpretation. This thesis aims to address some of these limitations through the help of machine learning.

The first chapter of this thesis provides an overview of molecular dynamics and current developments in force fields. It also introduces fundamental machine learning concepts, discusses popular machine learning potentials, and provides an overview of enhanced sampling techniques.

The second chapter presents different approaches to enhance the accuracy of force fields by combining machine learning, accurate long-range interactions, and nuclear quantum effects. We will introduce Deep-HP, a scalable multi-GPU neural network potential platform that has been integrated into the Tinker-HP package. Deep-HP enables the integration of neural network potentials with force fields. Specifically designed for carrying out large-scale Molecular Dynamics simulations, it offers enhanced capabilities for simulating large-scale molecular dynamics. This platform has been used to develop hybrid neural network polarizable force fields and has recently been combined with reactive physics-based neural network potential models, thus expanding its application to a wide range of research areas.

Lastly, the third chapter focuses on the development of various enhanced sampling techniques and their application in studying biomolecular complexes ranging from the discovery of SARS-CoV2 Main Protease (Mpro) inhibitors to protein binding. These techniques incorporate concepts from unsupervised machine learning and their efficiency was demonstrated their effectiveness in efficiently exploring conformational space. Additionnally, clustering and deep learning-driven Markov state processes were employed to analyze extensive data and elucidate the binding mechanisms of inhibitors within the SARS-CoV2 Mpro catalytic cavity, providing insights into the dynamics of ligand-protein binding modes.

Overall, this thesis presents a comprehensive analysis of how machine learning can enhance large-scale molecular dynamics simulations. It addresses the limitations of current approaches and highlights new perspectives for future research in this field.

Acknowledgement

I would like to sincerely thank Prof. Jean-Philip Piquemal. I am deeply thankful for the freedom I had during my Ph.D. research, which not only allowed me to shape my creativity but also helped me to grow as a researcher. I would also like to express my gratitude to the entire team. Olivier, the man who talks to computers, is one of the kindest persons I know and is always here for you. Pier, my coffee/beer buddy, played a crucial role in my personal and scientific growth in many ways. Thomas, another fervent coffee/beer buddy, in addition to being a friend, taught me so much, and without him, some of the work in this thesis would not have been possible. Diata, my supportive neighbor, whose constant smile always cheers me up. Frédéric, who stood by me during some of my research and was always there for me. Finally, Louis, for his invaluable assistance and for sharing memorable moments.

I also wish to express special gratitude to Prof. Pengyu Ren for the warm welcome I received at the University of Texas at Austin. It has been an honor and a pleasure to work with him and the entire Computational Biomolecular Engineering Lab, especially Yanxing and Chengwen, for their insightful discussions and collaboration, and also Elizabeth and Ketsia, who made my experience in Austin truly amazing and memorable.

I am truly thankful to my Ph.D. advisor, Prof. Gabriel Stoltz, Prof. Nohad Gresh for his scientific knowledge and engaging discussions, Prof. Pierre Monmarche for his mathematical insights, and Prof. Yvon Maday for the great collaboration with Gong.

I am also deeply grateful to my friends at the Laboratoire de Chimie Theorique and beyond, whether currently in Ph.D. or elsewhere, who have been so supportive and provided me so much strength and encouragement: Trini, Chiara, Timothee, Stefano, Umberto, Cesar, Igor, Iohanna, Jessica, Federica, Dina, Quentin, Gong, Daniele, David, Nicolai, Adam, and Igor.

These years have been more than just a Ph.D. journey. During this period, I had the pleasure of meeting incredible people who have now become friends. I feel extremely fortunate to have no negative memories throughout these years. While it's challenging to list everyone I've met, as I wouldn't want to unintentionally omit someone, I want to express my gratitude to all of you for the wonderful trips, shared moments, and the many more to come. I would also like to give special thanks to all my friends for their support throughout these years, especially Valentin, Jesus, Clement, Fabien, and Duo. The same goes for my family Sylvaine, Robert, Michele, Jean-Marc, Catherine and Augustin. Thank you also to Muriel and Nicolas for teaching me so much and for supporting me in

many ways.

Finally, I would like to dedicate this thesis to three of the people I love the most. Firstly, my mother and grandmother for their support throughout these years, giving me a strong and free mindset, and for all the love you gave me during all these years. Lastly, I want to thank my partner, Lucie, for being part of my life for the past 6 years, for standing by me during both bad and good moments. Without you, all of this would not have been possible, and it would not have tasted the same.

Contents

Introduction

Understanding the behavior of molecules and matter at the microscopic level is at the core of physics and chemistry. The advancement of computer technology, computing hardware, algorithms, and high-performance computing has greatly contributed to the development of simulation methods. In the 1960s, pioneering work by Karplus, Verlet, and Rahman introduced classical simulations for molecular systems, opening the path for computer simulations of molecules and biomolecules.[1, 2, 3] However, molecular dynamics (MD) simulations, which have become an essential tool for predicting the behavior of proteins in complex environments, face certain limitations.

The sizes of biological systems make it impossible to employ electronic structure methods, leading to the use of fitted empirical models known as force fields (FFs). Classical FFs are modeled by simplified formulas, e.g harmonic potential or truncated morse potential, making them computationally efficient. However, sometimes, it comes at the expense of accuracy. While offering an acceptable precision, they lack an accurate description of polarization and to a larger extent of all many-body physical effects which play a crucial in proteins structures. While the development of polarizable force fields (PFFs) has provided new insights into many-body effects and a better understanding of complex systems, [4, 5] they still rely on formula which may not capture important quantum mechanical effects. Additionally, the timescales accessible during MD simulations are often shorter than the processes of interest, especially in biomolecular simulations, necessitating the development of enhanced sampling algorithms to climb up high energy barriers. In addition, the high-dimensional nature of large simulation trajectories poses challenges in their interpretation and for extracting valuable information that could be compared with experimental data such as transition rates between states and free energy profiles.

This thesis aims to address these limitations through the use and development of machine learning models. By integrating machine learning into computational methods, we can enhance accuracy, accelerate simulations, and gain deeper insights into molecular phenomena.
In recent years, the parallel computing power of Graphics Processing Units (GPUs) and the availability of large datasets have paved the way for new paradigms in machine learning algorithms. Training machine learning models with billions of parameters on increasingly large databases has become routine. Specifically, significant efforts have been dedicated to including machine learning in FF developments. These models, known as Machine Learning Potentials (MLPs), aim to offer a mathematical relationship between atomic positions and potential energy, potentially bridging the accuracy and speed gap between FFs and *ab-initio* models.

However, most MLP models assume a purely local functional form and tend to neglect or implicitly account for long-range effects.[19, 20] Accurately describing long-range interactions is crucial for simulating condensed-phase systems and characterizing the structure of large proteins or DNA structures. Additionally, most MLP models are trained on neutral molecules, often overlook Nuclear Quantum Effects (NQEs), and face challenges in accurately representing water interactions which are crucial for biomolecular systems. Furthermore, the integration of efficient MLP multi-GPU infrastructure within existing molecular dynamics packages has been a significant challenge. While new Python-based ML architectures and dedicated machine learning libraries, such as PyTorch, TensorFlow, Scikit-learn, and Keras, have created a large community of developers and users,[6, 7, 8, 9], most molecular dynamics codes like CHARMM, GROMACS, and Tinker-HP [10, 11, 12, 13] use predominantly compiled languages like Fortran or C++. Consequently, executing Python-based MLP codes and MD codes simultaneously in a GPU-resident strategy is not an easy task.

To tackle these issues, Chapter 2, specifically Section 2.2, focuses on the development of Deep-HP, a multi-GPU platform within Tinker-HP, which enables the construction of hybrid ML-based PFF models and ensures their GPU-residency. This platform overcomes the limitations of host-device data transfers by transferring generic memory addresses through a Python/C++ interface and casting them into types compatible with MLP Python-based codes. The platform's capabilities are demonstrated through simulations of large biologically-relevant systems, such as the SARS-CoV-2 Mpro and Spike protein, using hybrid ML-based PFF models on hundreds of GPUs.

Building upon the Deep-HP platform, Chapter 2 Section 2.1 introduces the DNN-MBD model, which incorporates a Deep Neural Network (DNN) model within a linear scaling stochastic Many-Body Dispersion (MBD) model [14] to circumvent computationally expensive electron density calculations. A machine learning model is trained on local atomic properties, which are then used as input to compute long-range dispersion interactions. This approach not only avoids the need for explicit electron density calculations but also enhances the transferability of the model to larger and more complex systems. Additionally, its general framework allows its combination with a variety of methods from Density Functional Theory (DFT) functionals to FFs but also MLP models.

Another approach explored in this thesis involves using MLP to enhance the accuracy of existing PFFs. Chapter 2 Section 2.2 present a hybrid model that combines the ANI-2X MLP with AMOEBA in an embedding framework similar to QM/MM scheme. This hybrid model retains the highly accurate long-range effects of AMOEBA. The integration of multi-time-step integration techniques further accelerates simulations, enabling accurate free energy computations of 70 molecules in various solvents and the determination of binding free energies for 14 challenging host-guest systems from the SAMPL host-guest binding competitions.

In Section 2.4 we introduces as a perspective the Q-AMOEBA-NN model, a general machine learning-driven AMOEBA model based on an equivariant neural networks architecture. This model is also specifically designed to account for Nuclear Quantum Effects (NQEs) with the help of the adQTB model. The Q-AMOEBA-NN model's development is made possible through the AMOEBA parametrization of a vast database containing millions of conformations, including dipeptides, dimers, water clusters, and solvated ions.

The final Chapter of this thesis presents developments of efficient sampling algorithms, aiming to enhance the exploration of the conformational space of complex molecular systems. One of the algorithms, described in Chapter 3 Section 3.1, is the Adaptive Sampling (AS) technique. AS operates through iterations of parallel independent molecular dynamics replicas. The selection process of the AS algorithm relies on a low-dimensional projection of the molecule's Cartesian coordinates, obtained through various dimensionality reduction machine learning models, e.g autoencoders. The effectiveness of the AS algorithm was demonstrated by sampling the conformational space of the SARS-CoV-2 M^{pro}, resulting in the generation of 50 μs of simulations.
An improved version of the AS algorithm is presented in Chapter 3 Section 3.3, combining it with a novel generalized Gaussian-accelerated molecular dynamics (GAMD) method tailored specifically for PFFs and multi time-step integrators. This strategy achieved a speed factor of 15 in converging the free energy profile of large proteins, demonstrating its effectiveness in enhancing sampling efficiency. We explored how machine learning can enhance model accuracy and accelerate the exploration of the conformational space, enabling the production of long trajectories. However, an equally important aspect addressed in Sections 3.2 and B of Chapter 3 is the analysis of production data using machine learning techniques. Various clustering algorithms and deep learning-driven hidden Markov state models were employed in this thesis to extract biologically relevant structural and dynamical features and gain insights into critical structural behaviors. Through these analyses, valuable information regarding drug binding mechanisms and allosteric interactions, which are typically challenging to capture, could be deciphered.

The research conducted in this thesis aims to demonstrate the diverse applications of machine learning in addressing global challenges in molecular modeling. It encompasses endeavors to improve the accuracy of force fields, enhance sampling techniques, and gain valuable insights into biological molecular phenomena. The primary objective of this thesis is to pave new pathways in molecular modeling by harnessing the power of machine learning models, statistical mechanics, and parallel GPU computing. Through the ongoing development and future refinement of the Q-AMOEBA-NN model, along with the promising results obtained during this thesis, we hope that quantum-accurate simulations of million-atom systems will be possible in the near future.

1 State of the Art in Molecular Modelling

In this chapter, we provide an overview of the concepts and methods that will be used in this thesis. Firstly, we introduce the theoretical concepts in quantum chemistry, which are the foundation of numerous methods and serve as reference computations.
Next, in the second section, we discuss the limitations of these methods when applied to large systems, leading to introduce force fields and molecular dynamics. We present various ways to improve the accuracy of force fields and enhance the speed of molecular dynamics simulations.
In the third section, we explain general concepts of machine learning and provide an overview of machine learning potentials, including their construction and application, in section four.
In the final section, we focus on enhanced sampling techniques, highlighting how important they are for exploring rare events and how they can be combine with machine learning algorithms.

1.1 Quantum Chemistry

In this Section, we present the theoretical framework of quantum chemistry and discuss the methods employed in this study. We first introduce various quantum chemistry methods, ranging from wave function to density functional theory, which will used as reference benchmarks in this thesis for evaluating the accuracy of our models. These methods have been also employed to generate the datasets used for training machine learning models.

1.1.1 The Schrödinger Equation

Quantum mechanics is considered the most successful theory ever produced. Born in early 1900s through multiple Nobel Prizes ranging from Max Planck to Albert Einstein and passing by Louis de Broglie it has revolutionized many fields. Chemistry, for example, has particularly benefited from it since a proper description of electrons, responsible for the formation of chemical bonds, requires a quantum mechanical treatment.[21, 22, 23]

For simplicity, we will focus on the non-relativistic formulation of quantum mechanics. The non-relativistic electronic time-dependent Schrödinger equation describes the time evolution of a N_e electron and N atom system.

$$i\hbar\frac{d}{dt}\left|\Psi(\mathbf{X},t)\right\rangle = \hat{H}\left|\Psi(\mathbf{X},t)\right\rangle \tag{1.1}$$

Here, $\mathbf{X} = (\mathbf{R_1},\ldots,\mathbf{R_N},(\mathbf{r_1},\sigma_1),\ldots,(\mathbf{r_{N_e}},\sigma_{N_e}))$, are the space-spin coordinates with $\mathbf{r_i}$, and σ_i respectively the coordinates and spin of electron i, $\mathbf{R_j}$ are the coordinates of nulei j and $\Psi(\mathbf{X},t)$ is the wave function of the system at a given time t. Solving this equation consists in finding the eigenvalues and eigenvectors of the Hamiltonian operator $\hat{H} = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial z^2} + V(\mathbf{X},t)$. By assuming that $\hat{H}$ is time-independent, one can use the stationary states $\Phi(\mathbf{X})$ of the time-independent Schrödinger equation 1.2 to obtain the time-dependent solution.

$$\hat{H}\left|\Phi(\mathbf{X})\right\rangle = E\left|\Phi(\mathbf{X})\right\rangle \tag{1.2}$$

The time evolution of the stationary states is then simply obtained by multiplying them with a time-dependent phase factor.

$$\Psi_n(\mathbf{X},t) = \Phi_n(\mathbf{X})e^{\frac{iE_n}{\hbar}t} \tag{1.3}$$

With $n \in \mathbb{N}$ and E_n the energy associated to Φ_n. Since the Φ_n form a complete orthonormal basis, the initial wave function can be written as a linear combination of the Φ_n.

$$\Psi(\mathbf{X},0) = \sum_{n=0}^{\infty} c_n\Phi_n(\mathbf{X}) \tag{1.4}$$

Using the linearity of the Schrödinger equation, the time-dependent solution becomes

$$\Psi(\mathbf{X},t) = \sum_{n=0}^{\infty} c_n\Phi_n(\mathbf{X})e^{\frac{iE_n}{\hbar}t} \tag{1.5}$$

However, the computational time and memory required to solve this equation increases exponentially with the number of electrons. Thus solving this equation exactly is often impractical. Therefore, computational chemistry aims to develop methods that approximate its solutions. One such approximation, the Born-Oppenheimer (BO) approximation [24] neglects the kinetic energy of the nuclei, treating them as fixed point charges. With this assumption, the wave function of the system can be expressed as a product of a nuclear and an electronic wave function, $\Psi(\mathbf{R},\mathbf{Z}) = \Psi(\mathbf{R})\Phi(\mathbf{Z})$, with $\mathbf{R} = (\mathbf{R_1},\ldots,\mathbf{R_N})$ the nuclear coordinates and $\mathbf{Z} = ((\mathbf{r_1},\sigma_1),\ldots,(\mathbf{r_{N_e}},\sigma_{N_e}))$ are the space-spin electron coordinates. Notice that $\Phi(\mathbf{Z})$ is the electronic wavefunction but for a given positions of nuclear coordinates $\mathbf{R}$. This separation greatly simplifies the problem, making it more computationally efficient. This separation is at the basis of many quantum chemistry methods and force fields. It is motivated by the fact that both approaches are mainly used to study the ground state or the electronic structure of a system.

1.1.2 Wave-Function Methods

Within the framework of the Born-Oppenheimer (BO) approximation, the Hartree-Fock (HF) method [25] approximates the wave function as a single Slater determinant. This determinant is formed by antisymmetrizing a product of one-electron Spin-Orbitals (SOs). For the electronic degrees of freedom, this description implies that each electron experiences the mean field created by all other electrons. While the HF method provides nearly 99% of the total energy, it cannot fully capture the electronic correlation, which is essential for describing molecular systems accurately. When we refer to accuracy, we specifically mean achieving chemical accuracy, which is the precision necessary for making realistic chemical predictions, typically below 1 kcal/mol. More advanced methods, such as post-HF methods, start with HF SOs and use strategies to consider the exact Coulomb repulsion interaction to recover the correlation energy. These methods are required to accurately describe molecular systems and are constantly being developed and refined.

One of the first post-HF methods, Møller–Plesset perturbation theory (MP) [26], adds electronic correlation via perturbation theory. Its truncation to the second order (MP2) is often used to calculate equilibrium geometries as it provides a good cost/accuracy trade off for medium-sized systems. MP2 equilibrium geometries are often considered as reference in quantum chemistry. For instance, the datasets employed in force field parametrization (discussed in Section 1.2.2) or in machine learning potentials (see Section 1.4) usually use MP2 geometries and perform higher-level energies and forces single-point computations.

One popular post-HF method is the Configuration Interaction (CI) which describes the wave function by a linear combination of Slater determinants, built from HF SOs, where the coefficients are determined by applying the variational method to minimize the energy. The full CI case (FCI), where all excited Slater determinants are included (single, double, ...), provides an exact solution to the time-independent non-relativistic Schrödinger equation.

$$|\Psi_{CI}(\mathbf{Z})\rangle = \sum_{n=1}^{\infty} C_n D_n(\mathbf{Z}) \tag{1.6}$$

Here, C_n are the CI coefficients, and $D_n(\mathbf{Z})$ are Slater determinant constructed from the HF SOs. The CI coefficients corresponding to the ground-state wave function are obtained by minimizing the CI energy.

$$C_n = \arg\min \frac{\langle \Psi_{CI}| \hat{H} |\Psi_{CI}\rangle}{\langle \Psi_{CI}|\Psi_{CI}\rangle} \tag{1.7}$$

However, the full-CI is often computationally expensive. Therefore, in practice, CI calculations are often truncated based on criteria such as orbital active space or excitation level. One common truncation approach is limiting the CI space to single and double excitations (determinants) resulting in the CISD method. Alternatively, one can employ iterative procedures or various stopping criteria to selectively retain the most relevant determinants, ensuring control over the quality of wave functions

and energy estimates. [27, 28, 29, 30, 31] One of the major shortcoming of truncated CI methods is their lack of size extensivity. The size extensivity can be defined as the total energy of a system composed of two non-interacting fragments A and B must be the sum of the total energies of the separate fragments in the limit of infinite separation.

$$E(A \dots B) = E(A) + E(B) \tag{1.8}$$

This property is especially important in chemistry as systems are often composed of fragments such as atoms, molecules or amino acids. The size-inconsistency of truncated CI has led to the development of the coupled-cluster (CC) theory, which is widely considered as one of the gold-standard method in quantum chemistry.

The Coupled Cluster (CC) method also uses HF SOs as starting point and constructs a wave function by using a parametrized cluster operator $\hat{T}$.

$$|\Psi_{\text{CC}}(\mathbf{Z})\rangle = e^{\hat{T}} |\Phi_{HF,0}(\mathbf{Z})\rangle \tag{1.9}$$

$|\Phi_0(\mathbf{Z})\rangle$ is the HF wave function. The cluster operator is the sum of cluster operators of different excitation levels, such as single, double, and higher-order excitations.

$$\hat{T} = \hat{T}_1 + \hat{T}_2 + \dots \tag{1.10}$$

To understand the action of the operator $e^{\hat{T}}$ on the HF wave function , one can expand the exponential through the use of Taylor expansion.

$$e^{T} = 1 + T_1 + T_2 + \frac{1}{2}T_1^2 + \frac{1}{2}T_1 T_2 \dots \tag{1.11}$$

If the cluster operator $\hat{T}$ is left untruncated, then the Full CC (FCC) wave function becomes only a nonlinear reparametrization of the FCI. However, the FCC as well as FCI is also computationally expensive. The real interest of the CC method arises when the cluster operator is truncated. First, the CCSD wave function includes much more excited determinants compared to the CISD wave function, while maintaining the same number of parameters. Additionally, a major advantage of truncated CC, is its size-extensivity, which directly stems from the exponential structure of the wave function. CCSD and especially with its perturbative inclusion to triple excitations CCSD(T) is considered as the "gold standard" of computational chemistry when extrapolated to the basis set limit (CCSD(T)/CBS, CBS for complete basis set) and is often used as reference to test models (see Chapter 2, Section 2.1) and for the generation of small molecule dataset (ANI dataset suit, see latter in Section 1.4.3). However, it is computationally expensive and does not scale well with the number of electrons $\mathcal{O}(N_e^7)$. As a result, its use to model biologically relevant systems is impractical. While there have been attempts to improve its scaling, such as with the linear scaling domain based local pair-natural orbital singles and doubles coupled cluster method, DLPNO-CCSD(T), this approach is still computationally

expensive for biomolecular systems.[32]

1.1.3 Density Functional Theory

Density Functional Theory (DFT) [33] provides a useful alternative to wave function methods by demonstrating that the electronic energy of a system can be uniquely determined by its electron density $\rho(\mathbf{r})$. This greatly reduces the many-body N_e electrons problem from $3N_e$ coordinates to just three, allowing for the simulation of larger systems ranging from hundreds to thousands of atoms. DFT breaks down the electronic energy into several contributions and aims to find the ground-state electronic energy by minimizing a functional that depends on the electron density.

$$E[\rho(\mathbf{r})] = T_{\mathrm{s}}[\rho(\mathbf{r})] + E_{\mathrm{xc}}[\rho(\mathbf{r})] + J[\rho(\mathbf{r})] + E_{\mathrm{ne}}[\rho(\mathbf{r})] \tag{1.12}$$

Here, $E_{\mathrm{xc}}[\rho(\mathbf{r})]$ is the exchange-correlation energy, $E_{\mathrm{ne}}[\rho(\mathbf{r})]$ the nuclei-electron interaction energy, $T_{\mathrm{s}}[\rho(\mathbf{r})]$ the non-interacting kinetic energy and $J[\rho(\mathbf{r})]$ the electron-electron repulsion energy. However, the major problem with DFT is that the exact form of the exchange-correlation functional is not known, except for the free-electron gas case. Thus, a variety of DFT approximations have been proposed, leading to increasingly better performance across a broad range of properties from chemistry to material science.

We will start by the Local Density Approximation (LDA) [34] class of functionals. These functionals are based on the assumption that the electron density changes slowly and can be treated locally as a uniform electron gas (UEG).

$$E_{\mathrm{xc}}^{\mathrm{LDA}}[\rho] = \int_{\mathbb{R}^3} e_{\mathrm{xc}}^{\mathrm{UEG}}(\rho(\mathbf{r}))d\mathbf{r} \tag{1.13}$$

$e_{\mathrm{xc}}^{\mathrm{UEG}}(\rho(\mathbf{r}))$ is the exchange-correlation potential density of the UEG. In this case, at any given point in the space $\mathbf{r}$, only the local density $\rho(\mathbf{r})$ needs to be computed in order to determine the value of the exchange-correlation functional. While this locality assumption is relatively crude, the LDA functional has proven to be successful in qualitatively describing various properties especially for solids where the electron-gas assumption is often acceptable. However, LDA shows some limitations in systems where the electron density is not uniform such as in molecules.

The next step beyond the LDA is the Generalized Gradient Approximations (GGA). The GGA exchange-correlation functional $E_{\mathrm{xc}}^{\mathrm{GGA}}[\rho]$ is not only a function of the local value of the electron density $\rho(\mathbf{r})$ but also its gradient $\nabla\rho(\mathbf{r})$.

$$E_{\mathrm{xc}}^{\mathrm{GGA}}[\rho] = \int_{\mathbb{R}^3} e_{\mathrm{xc}}^{\mathrm{GGA}}(\rho(\mathbf{r}), \nabla\rho(\mathbf{r}))d\mathbf{r} \tag{1.14}$$

GGA functionals provide an important improvement over LDA in most cases and many functionals have been proposed. The Perdew-Burke-Ernzerhof (PBE) [35] exchange-correlation is one of the most popular one. In comparison to the GGA functionals available at the time, the PBE exchange and correlation energies are simpler functions of $\rho(\mathbf{r})$ and $\nabla\rho(\mathbf{r})$, thereby imposing less conditions but more importantly it does not rely on fitted parameters, rendering it more *ab-initio* in nature.

Hybrid functionals, introduced by Becke [36], are another well-known type of functionals. They mix a fraction of HF exchange with part of GGA functionals to improve the calculation of molecular properties, such as bond lengths or atomization energies which were not accurately described by *ab-initio* functionals at that time. These functionals rely on parameters obtained through fitting to experimental data. The aim of adding a fraction of HF exchange is to alleviate the self-interaction error which tends to favor overly delocalized electron densities instead of localized ones. The most famous and widely used hybrid functional is B3LYP and used three parameters.[37] Another popular functional is PBE0 [38], which is based on a single parameter a and incorporates the PBE exchange correlation energy, making it much simpler to interpret.

$$E_{\mathrm{xc}}^{\mathrm{PBE0}}[\phi] = aE_{\mathrm{x}}^{\mathrm{HF}}[\rho] + (1-a)E_{\mathrm{x}}^{\mathrm{PBE}}[\rho,\phi] + E_{\mathrm{c}}^{\mathrm{PBE}}[\rho,\phi] \tag{1.15}$$

While many GGA and hybrid functionals have been developed in the past years, in the following we focus exclusively on PBE and PBE0 as they are widely recognized as standard methods. This choice enables us to make direct comparisons with results in the literature and is the main reason why we coupled our DNN-MBD model in Chapter 2, Section 2.1, with both of them.

In principle, DFT is exact, which implies that the true functional should incorporate dispersion interactions. However, in practice, accurately capturing dispersion interactions within current DFT frameworks is challenging. Dispersion interactions are long-range correlation effects that are inherently difficult to account accurately. Determining suitable expressions for long-range correlation that maintain the balance between exchange and correlation is difficult.

1.1.4 Dispersion corrections

Dispersion interactions are a crucial components of non-covalent interactions, but as explained before they present a challenge for DFT methods. Indeed, while these functionals can describe many properties, they struggle with long range interactions such as charge-transfer as only local contributions to electronic correlation are included. Range-separated functionals, such as the ωB97X exchange-correlation functional, can partially overcome this limitation by splitting the electron-electron repulsion into short and long-range components.

$$E_{\mathrm{xc}}^{\omega\mathrm{B97X}}[\phi] = aE_{\mathrm{xc}}^{\mathrm{sr,HF}}[\phi] + (1-a)E_{\mathrm{xc}}^{\mathrm{sr,PBE}}[\rho] + E_{\mathrm{xc}}^{\mathrm{lr,PBE}}[\rho] + E_{\mathrm{c}}^{\mathrm{PBE}}[\rho,\phi] \tag{1.16}$$

However, none of the exchange-correlation approximations can fully describe London dispersion interactions, which are essential for modeling intermolecular interactions. To address this limitation, semi-empirical pairwise dispersion correction terms can be added to standard DFT, where the dispersion coefficients for atom pairs or triplets are tabulated.[39, 40, 41] The semi-empirical dispersion energy can be simply represented by the attractive component in Lennard-Jones potential.

$$E_{disp} = \sum_{i=1}^{N} \sum_{i'>i}^{N} f(R_{ii'}, R_i^{\mathrm{VdW}}, R_{i'}^{\mathrm{VdW}}) \frac{C_{ii',6}}{R_{ii'}^6} \tag{1.17}$$

Where R_i^{VdW}, $R_{i'}^{\mathrm{VdW}}$ are atomic VdW radii, $R_{ii'}$ is the distance between the atom pairs and $C_{ii',6}$ pairwise coefficients. Other, less empirical dispersion model have been proposed such as the Tkatchenko and Scheffler (TS) scheme where the pairwise C_6 coefficients are instead expressed in terms on free atom reference data and atom-in-molecule (AIM) polarizabilities.[42] These polarizabilities are obtained through the AIM partitioning of the electron density.[43] While these methods are computationally cheap, compared to the solution of DFT, they still rely on tabulated values and fail to capture the many-body nature of dispersion interactions.

To address this limitation, Tkatchenko et al. have proposed a range-separated many-body dispersion model where dispersion is calculated as the long range correlation energy of interacting oscillating dipoles, centered on the atomic positions.[17] While relying only on a single parameter it uses a diagonalization procedure which results in a cubic scaling with the number of atoms N, $\mathcal{O}(N^3)$. Recently, Poier et al. used a stochastic Lanczos trace estimator to circumvent the diagonalization procedure opening a new path of linear scaling dispersion-corrected DFT simulations.[44, 14] In Chapter 2 Section 2.1, we will discuss how neural networks can be coupled with this model thus enabling its inclusion within force fields development.

The methods described earlier allow to compute electronic structures, but for studying the dynamics of a system and capturing ensemble properties required for most experimental observables, sampling approaches are needed. One approach for sampling the configuration space of a system is through Molecular Dynamics (MD) simulations. MD simulations can be categorized into two types: *ab-initio* MD and classical MD. *ab-initio* MD uses energies and gradients obtained from electronic structure methods described in previous Sections. However, *ab-initio* MD simulations are computationally expensive and typically limited to systems with thousands of atoms. On the other hand, classical MD was developed to study large systems, more than million of atoms. While classical MD neglects the dynamics of the electrons and lacks quantum mechanical phenomena, making it unsuitable for studying reactivity such as bond breaking, its lower computational cost allows for exploration of more complex systems.

In this thesis, electronic structure methods were used to develop a database aimed at improving classical MD models. However, before delving into that, let's first delve into classical MD.

1.2 Molecular Dynamics

As a major part of this thesis has focused on force field development, parametrization, and large-scale molecular dynamics simulations, this section begins by providing fundamental insights into these topics. Additionally, we present acceleration techniques, as intensive work has been done on high-performance computing simulations and GPU acceleration. Furthermore, we introduce some concepts in nuclear quantum effects, which are currently in used in the coupling of our machine learning-powered force field models, Q-AMOEBA-NN.

1.2.1 Basics of Molecular Dynamics

Understanding the dynamics of biological systems is crucial in structural biology and drug discovery. Over the past five decades, classical MD simulations have become a vital theoretical tool for predicting the long-time behavior of proteins in complex environments. This approach is particularly useful for predicting the complete conformational space of proteins beyond just their simple structure.

To achieve this, the simulation must be ergodic, meaning that an infinitely long trajectory can fully describe its statistical properties. In other words, all accessible microstates, defined as possible arrangements of molecular position at a particular thermodynamic state, must be equiprobable over a long timescale. This ensure that any observable $\mathcal{A}$, such as thermodynamic quantities, can be expressed as a time-average of this observable. In the following we will denote by $\mathbf{r}(t)$ the Cartesian coordinates of the atoms at time t during a dynamics while $\mathbf{P}(t)$ will denote their corresponding momenta.

$$< \mathcal{A} >= \lim_{t' \longrightarrow \infty} \frac{1}{t'} \int_0^{t'} \mathcal{A}(\mathbf{r}(t), \mathbf{P}(t)) \mathrm{d}t \approx \frac{1}{t'} \sum_{t=1}^{t'} \mathcal{A}(\mathbf{r}(t), \mathbf{P}(t)) \tag{1.18}$$

To compute it, a statistical ensemble must be defined in which the simulations will be performed. Each statistical ensemble conserves specific properties during the simulation. For example, the microcanonical NVE ensemble maintains a constant number of particles N, volume of the simulation box V and total energy of the system E. The canonical ensemble NVT fixes, N, V and T, the temperature of the simulation. Finally, the isobaric ensemble NPT fixes N, P the pressure and T. In biochemistry simulations, the isobaric ensemble is often used as the temperature and pressure are typically constant (atmospheric pressure), and for human proteins, the temperature is set to the human body temperature.

The MD framework, starts by using the BO approximation. In the case of all-atoms simulations, the time evolution of the nuclei is obtained through Newton's equations of motions. This is done by discretizing the time into δt time-step.

$$\mathbf{m}_i \mathbf{a}_i = \mathbf{m}_i \frac{\partial \mathbf{v}_i}{\partial t} = -\frac{\partial E}{\partial \mathbf{r_i}} \tag{1.19}$$

Considering $\mathbf{m}_i$ as the mass of atom i, $\mathbf{a}_i$ as its acceleration, $\mathbf{v}_i$ as its velocity and $-\frac{\partial E}{\partial \mathbf{r}_i}$ as the forces acting this atom. The goal is to determine the positions and velocities at time $t + \delta t$ based on the positions and velocities at time t, e.g $\mathbf{r}_i(t)$ and $\mathbf{v}_i(t)$. To achieve this, one of the most famous numerical method used to integrate Newton's equations of motion is the Velocity-Verlet algorithm [2, 45]. This algorithm is numerical stable, error of $\mathcal{O}(\delta t^3)$ for both velocities and positions, while ensuring time-reversibility of the dynamics. Later in the thesis, Chapter 2, we will discuss about multi time-step integrators which is another class of integrators that evaluate different part of a potential at different time-step.

If the size of the simulation box is fixed, solving Newton's equations of motion is equivalent to conducting simulations in the microcanonical ensemble NVE. However, if one intends to simulate systems in other ensembles, it becomes necessary to control the temperature or the pressure throughout the dynamics. Various thermostat algorithms have been proposed such as the Nose-Hoover thermostat, the Langevin thermostat, and the Berendsen thermostat.[46, 47] To conduct simulation in the isobaric ensemble (NPT), a barostat is required to control the pressure by scaling the simulation box. However, accurately computing the virial, necessary for pressure control, is more challenging than computing instantaneous temperature. Various barostat methods have been developed, such as Berendsen, Bussi, Monte Carlo, or more recently the Martyna-Tuckerman-Tobias-Klein barostats [48, 49, 50, 51]. All the simulations conducted in this thesis used the Monte Carlo barostat which considered as standard robust barostat. Even, if the MTTK barostat is currently considered the most efficient barostat.

In the next section we will discuss about empirically parametrized potentials, known as force fields, used in classical MD.

1.2.2 Force Fields

Force fields are cheap and scalable potentials that are usually used to simulate very large systems such as the ones in biology. But they are not restricted to biology as they are also used to simulate polymers and complex materials.
Force fields are empirical models based on relatively simple mathematical formula such as harmonic functions, sum of cosine or truncated Morse potential that makes them computationally extremely efficient. These formula depends on parameters that are atom class dependent. An atom class can be seen as a specific atomic environment, for instance if the atom is within a specific amino acids or organic molecule. These parameters are optimized on a dataset that can be composed of experimental as well as highly accurate quantum mechanical data. The parametrization of force fields will be thoroughly discussed in Section 1.2.2. Common examples of FFs are OPLS,[52] AMBER,[53] CHARMM,[54] GROMACS,[11] and AMOEBA.[55, 56, 57]
The potential energy functional of FF can usually be separated into bonded and non-bonded terms.

$$E_{\text{FF}} = E_{\text{bonded}} + E_{\text{non-bonded}} \tag{1.20}$$

The role of the bonded term is to capture the energy fluctuations due to the distorsions of the molecular geometry. Its functional form varies between FFs but usually includes bond-stretching, angle-bending, and dihedral-torsion interactions.

$$E_{\text{bonded}} = E_{\text{bond}} + E_{\text{angle}} + E_{\text{torsion}} + \dots \tag{1.21}$$

Typically, harmonic potentials are used to model bond-stretching and angle-bending, with equilibrium bond length R_0 and angle θ_0, stiffness constants k_{bond} and k_{angle}.

$$E_{\text{bond}} = \sum_{i,j}^{\text{bonds}} k_{ij}(\mathbf{r_{ij}} - R_0)^2 \tag{1.22}$$

$$E_{\text{angle}} = \sum_{i,j,k}^{\text{angles}} k_{ij}(\theta_{\mathbf{ijk}} - \theta_0)^2 \tag{1.23}$$

These parameters are optimized by quantum mechanical consideration such as diatomic distances and experimental bond frequencies. Despite its simplicity, the harmonic approximation works well, as most stretching displacements remain close to equilibrium in biomolecular simulations.
On the other hand, torsions involve atom rotations around an axis and cannot be modeled using the harmonic approximation as this potential is much smoother and due to the periodicity characterizing torsional degrees of freedom. A truncated Fourier series is commonly used to represent the torsion potential. The coefficients of the Fourier series V_n, modulate the height of the energy barrier of the given torsion angle. The degree of truncation n_{max}, linked to the degree of freedom, is usually set to 3.

$$E_{\text{torsion}} = \sum_{n=1}^{n_{\text{max}}} V_n \cos(n\omega) \tag{1.24}$$

There are slight variations in the bonded functional form among FFs, particularly when considering the AMOEBA polarizable force fields, which will be extensively discussed in the following sections as it was the basis of many method developments in this thesis. The bonded term of AMOEBA incorporates additional components that effectively modulate a wide spectrum of physical phenomena.

$$E_{\text{bonded}} = E_{\text{bond}} + E_{\text{angle}} + E_{\text{torsion}} + E_{\text{oop}} + E_{\text{stretch-bend}} \tag{1.25}$$

The bond-stretching and angle-bending interactions are modeled using a fourth-order Taylor expansion of the Morse potential. The torsion interactions are represented by a truncated Fourier series.

$$E_{\text{bond}} = \sum_{ij}^{\text{bonds}} k_{\text{bond}}(\mathbf{r_{ij}} - R_0)^2(1 - 2.55(\mathbf{r_{ij}} - R_0) + 3.793125(\mathbf{r_{ij}} - R_0)^2) \tag{1.26}$$

$$E_{\text{angle}} = \sum_{ijk}^{\text{angles}} k_{angle}(\theta_{\mathbf{ijk}} - \theta_0)^2(1 - 0.014(\theta_{\mathbf{ijk}} - \theta_0) + 5.6 \times 10^{-5}(\theta_{\mathbf{ijk}} - \theta_0)^2) \tag{1.27}$$

$$E_{\text{torsion}} = \sum^{\text{torsions}} \sum_{n=1}^{n_{\text{max}}} V_n(1 + \cos(n\omega - \Omega_{\text{torsion}}) \tag{1.28}$$

To maintain the planarity of specific functional groups, an out-of-plane bending (oop) interaction term is incorporated and modeled using the Wilson-Decius cross function, while the stretch-bend term is taken from the MM3 force field. [58]

$$E_{\text{oop}} = \sum_{\text{oop}} a k_{\text{oop}} \chi^2 \tag{1.29}$$

$$E_{\text{stretch-bend}} = \sum_{ijk}^{\text{str-bends}} k_{\text{str-bend}}((\mathbf{r_{ij}} - R_0) + (\mathbf{r_{jk}} - R_0))(\theta_{\mathbf{ijk}} - \theta_0) \tag{1.30}$$

While this show that there are variations in how force fields model short-range interactions are described, the non-bonded term, which describes interactions between multiple molecules, is more complex and is at the core of the diversity in FFs.
A majority of FFs incorporate a van der Waals (VdW) term to handle both short-range repulsion and dispersion interactions usually through a 12-6 Lennard-Jones potential.

$$E_{\text{VdW}} = \sum_{ij} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{\mathbf{r_{ij}}} \right)^{12} - \left(\frac{\sigma_{ij}}{\mathbf{r_{ij}}} \right)^{6} \right] \tag{1.31}$$

Here ϵ_{ij} is the depth of the pair potential and σ_{ij} the distance at which the potential is set to zero.
The $\frac{1}{R^6}$ term is related to the pairwise dispersion interactions equation 1.17. The $\frac{1}{R^{12}}$ is used to approximate short-range Pauli repulsion, and it is empirically grounded as the square of the dispersion interaction $\frac{1}{R^6}$.
The AMOEBA polarizable FF uses a modified version of the Lennard-Jones potential, called the buffered 14-7 equation, to improve fitting to gas-phase *ab-initio* and liquid-phase properties of noble gases.

$$E_{\text{VdW}} = \sum_{i,j} \epsilon_{ij} \left(\frac{1.07}{\frac{\mathbf{r_{ij}}}{R^0_{ij}} + 0.07} \right)^7 \left(\frac{1.12}{\left(\frac{\mathbf{r_{ij}}}{R^0_{ij}} \right)^7 + 0.12} \right) \tag{1.32}$$

where R^0_{ij} is the minimum energy distance and ϵ_{ij} the potential energy minimum that are use combining rules based on atomic parameters.
The electrostatic interactions are even more FF dependent and are at the core of differences between FF families such as classical, e.g AMBER and OPLS or polarizable, e.g AMOEBA and CHARMM, ones. In the following we will discuss how electrostatic interactions are modelled in these two families of FFs.

Non Polarizable Force Fields

There is a wide range of non polarizable force fields, each with their own unique parametrization procedure and models. The Universal Force Field (UFF) is an example of a force field that covers all elements in the periodic table, including actinides.[59] On the other hand, the Consistent Force Field (CFF) focuses more on organic compounds, polymers, and metals.[60] Additionally, non nolarizable

force fields like CHARMM and AMBER are designed to simulate biochemical systems such as nucleic acids and proteins.[54, 61, 53] Meanwhile, the Optimized Potentials for Liquid Simulations (OPLS) force field is specifically tailored to reproduce liquid phase properties.[52]
In most of non nolarizable FF, the electrostatic interactions employ a simplistic approach where atomic point charges are fixed at the center of the on atoms and interact with each other via Coulomb's law

$$E_{\text{electrostatic}} = \frac{q_i q_j}{\mathbf{r_{ij}}} \tag{1.33}$$

While this approach is computationally cheap and quite effective in describing the potential energy surface (PES), it neglects the response of the electron to changes in the environment. Therefore, to account for this many-body polarization effect, an additional term must be incorporated into the FF.

Polarizable Force Fields

Polarizable force fields (PFFs) are designed to incorporate polarization, i.e the response of electron density to electrostatic intra- and intermolecular perturbations, going beyond the typical electrostatic description based on fixed atomic point charges.
The inclusion of polarization is accomplished by different approaches.

The Drude oscillator model

The Drude oscillator model partitions the atomic partial charge into a nuclear and fictitious mass-less component, known as the Drude particle, connected through a harmonic potential, with the sum of the charges being equal to the reference atomic charge. The interaction between the fictitious charge on an atom and all other charges results in an atomic induced dipole.
In the Drude model, the energy can be decomposed into three terms,

$$E_{\text{Drude}} = \frac{1}{2}\sum_{i=1}^{N} k_{\text{Drude}}(R_{D,i} - \mathbf{r_i})^2 + \sum_{j=1}^{N_D}\sum_{i=1}^{N} \frac{q_{D,j} q_i}{|R_{D,j} - \mathbf{r_i}|} + \sum_{i,j=1}^{N_D} \frac{q_{D,j} q_D(i)}{|R_{D,j} - R_{D,i}|} \tag{1.34}$$

where the N_D is the total number of Drude particles, $R_{D,i}$ its position according to atom i and $q_{D,i}$ its partial charge. k_{Drude} is the stiffness constant of the harmonic oscillator between the Drude particle and atom i.
The placement of the Drude particles R_D should be done self-consistently to find their energy minimum, which is computationally demanding. To reduce computational costs, extended Lagrangian techniques are used by assigning a small fictitious mass to each Drude particle. In this scheme, the mass of each relevant atom is divided between the parent atom it is connected to and its corresponding fictitious particle. However, if the mass allocated to the fictitious particle is not carefully chosen it may induce either small time steps or will thereby violating the Born-Oppenheimer approximation. Once the mass allocation is determined, the particles movements are calculated using standard integration methods.
However, this approach is not computationally inexpensive since it involves a greater number of

electrostatic interactions, through the computation of three terms, as shown in equation 1.34. Additionally, Drude oscillators approach cannot benefit from multi time-step integration techniques which have been extensively used to speeding up more advanced force fields. One of the famous FF that including them is CHARMM.[10, 61]

Point Dipole model

The polarizable dipole model incorporates polarization $E_{\text{elect,pol}}$ at the dipole level by incorporating atom-centered dipole polarizabilities.

$$E_{\text{elect,pol}} = \frac{1}{2}\mu^{\intercal}\mathbf{T}\mu - E^{\intercal}\mu \tag{1.35}$$

μ is a vector of dimension $3N_p$, with N_p the number of polarizable sites, that is composed of all induced dipole moment components. E is the electric field components arising from permanents moments, monopoles, dipoles and quadrupoles located at an atomic center and defined within a local coordinate system that is established by the atomic center and its surrounding bonded neighbors. The Distributed Multipole Analysis (DMA)[62] technique is used for the initial calculation of these multipoles. For further information we refer to the book of Stone.[63] Finally, the polarization matrix, $\mathbf{T}$, is taken into account to consider the interaction between dipoles μ_i and μ_j.

$$\mathbf{T}_{ij}^{\beta\gamma} = -\frac{\delta_{\beta\gamma}}{\mathbf{r}_{ij}^3} + 3\frac{\mathbf{r}_{ij}^{\beta}\mathbf{r}_{ij}^{\gamma}}{\mathbf{r}_{ij}^5} \tag{1.36}$$

In order to prevent the polarization catastrophe phenomenon, which arises from induced dipoles diverging when atoms are in close proximity, Thole proposed the use of damping functions known as Thole damping factors.[64]

Although for the sake of notation the Thole damping factors are omitted, the full polarization matrix $\mathbf{T}$ can be written using a 3×3 block matrix.

$$\mathbf{T}_{ij}^{\beta\gamma} = \begin{pmatrix} \alpha_1^{-1} & -\mathbf{T}_{1,2} & \cdots & -\mathbf{T}_{1,N_p} \\ -\mathbf{T}_{2,1} & \alpha_2^{-1} & \cdots & -\mathbf{T}_{2,N_p} \\ \vdots & \vdots & \ddots & \vdots \\ -\mathbf{T}_{N_p,1} & \mathbf{T}_{N_p,2} & \cdots & \alpha_{N_p}^{-1} \end{pmatrix} \tag{1.37}$$

With β, γ the component of the electric field experienced by the atom i. To better comprehend this equation, $\mathbf{T}_{ij}\vec{\mu}_j$ expresses the electric field created by dipole j on atom i, while $\mu_i^{\intercal}\mathbf{T}_{ij}\mu_j$ denotes the interaction energy between these two induced dipoles.

To compute a set of $3N_p$ induced dipoles μ, the energy in equation 1.35 must be minimized. Which is equivalent to solving the $3N_p \times 3N_p$ linear system.

$$\mathbf{T}\mu = E \tag{1.38}$$

Finally, plugging it in equation 1.35, simplify the polarization energy to

$$E_{\text{elect,pol}} = -\frac{1}{2} E^{\intercal} \mu \tag{1.39}$$

The whole problem is reduced of finding the solution of induced dipoles model μ by solving the linear equation of 1.38 at each time-step by inverting $\mathbf{T}$. Solving this equation can be done efficiently through the use of Krylov approaches such as the Preconditioned Conjugate Gradient for which its scalability and robustness has been discussed intensively in previous works.[65, 66]
The induced dipole model has been shown to be better suited for polarizable systems, such as ionic liquids, compared to Drude oscillators, resulting in higher accuracy.[67, 68]
AMOEBA [69] is a notable example of a polarizable FF that utilizes the point dipole model. The set of permanent atomic multipoles in AMOEBA includes charges, dipoles, and quadrupoles.
When studying liquid or solid systems, we are typically interested in their bulk properties. As such, periodic boundary conditions (PBC) must be applied. In the following section, we will explore how to compute these electrostatic interactions in PBC through the use of the Particle Mesh Ewald method.

Particle Mesh Ewald

When simulating pure water model or biomolecules in solution, surface effects can compromise drastically the accuracy of the simulation, as the molecules near the surface experience different forces than those in the center. One way to mitigate these effects is by adding more solvent molecules, but this is not practical due to the increased computational cost and due to the need of imposing constraints to prevent evaporation. An alternative approach is to use Periodic Boundary Conditions (PBC), where the system is placed inside a box and replicated in all three directions of the space to form a periodic lattice. This ensures that the solvent molecules near the box boundaries are no longer exposed to vacuum, but to the periodic image of the box. In addition, molecules leaving the central box are replaced by those from the neighboring box, eliminating the problem of evaporation. However, the calculation of forces and energy has a formal complexity of $\mathcal{O}(N^2)$. One way to circumvent this in the case of the electrostatic potential is to use Particule Mesh Ewald (PME).
While the vdW terms decay quickly in, at least, $\frac{1}{R^6}$, the electrostatic term have a slow decay in $\frac{1}{R}$ meaning that one should include the interactions from many periodic images. PME takes his origin through the Ewald summation. In the Ewald summation method, the electrostatics interactions are divided into absolutely converging sums, a direct, a reciprocal and a correction term:

$$E_{\text{electrostatic}} = E_{\text{direct}} + E_{\text{reciprocal}} + E_{\text{self}} \tag{1.40}$$

The splitting distance for the direct and reciprocal terms is controlled by a parameter. This reduce the scaling to $\mathcal{O}(N^{\frac{3}{2}})$.
PME improves the computation of the reciprocal term through a convolution product. The convolution product can be easily computed in the Fourier space by projecting the charges on a grid and through the

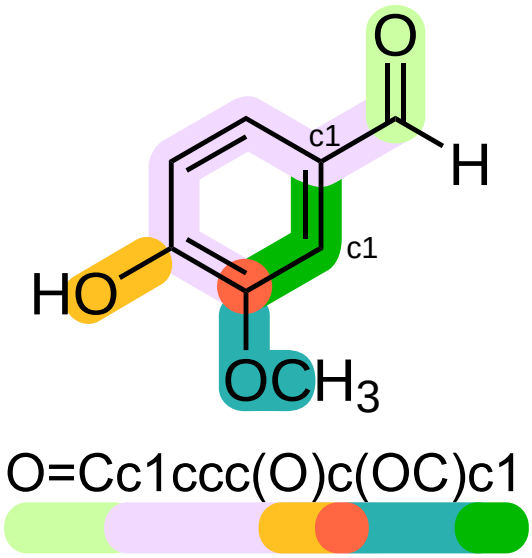


Fig. 1.1.: Example of a SMILES string, which is a compact and unique textual representation of a molecule's structure and is at the core of most of FF parametrization procedure.

use of Fourier transform. The use of fast Fourier transforms (FFTs) also accelerates the computation of $\mathbf{E}_{\text{reciprocal}}$ while reducing the complexity to $\mathcal{O}(Nlog(N))$.

Parametrization

One important question is how to parametrize a FF? This is a challenging task because the accuracy and transferability of a FF depend on the quality of its parameters. In this context, we focus on the parametrization procedure of the AMOEBA FF, which has been successfully applied to a wide range of molecular compounds, including small organic molecules, proteins, and nucleic acids. However, due to the continuous discovery of novel molecules, an automatic and rapid parametrization framework is required.

Poltype and its recent extension Poltype2 is a framework that automatically assign AMOEBA FF parameters to small molecules based on a pre-existing database of previous AMOEBA parametrization or through a fitting of *ab-initio* computations generated on the fly.[70, 71] The input chemical structure is provided to the program in the form of an SDF file, which contains information about the positions of atoms and their connectivity and hybridization state. The molecular structures are also encoded by a string called SMILES, as shown in Figure 1.1.

Poltype combines two databases of previously parametrized small organic molecules, AMOEBA09 and AMOEBA21, and employs a fragment database of SMILES string descriptions of fragments derived for torsion parameters. The program matches the SMILES string of the input molecule with the best fragment from the database to get the parameters.

FF parametrization uses atom classes (AC) and atom types (AT) to characterize a chemical environment. AT is a subset of AC, which makes AC less sensitive to local environment. Each parameter of the FF terms is either parametrized using AC or AT. For example, the bonded parameters, e.g k_{bond} and R_0, are AC-specific while non-bonded parameters use AT to provide greater flexibility.

Poltype first performs a substructure search on the input molecule to determine which atoms belong to the same AT or AC. This is done using an array of graph invariants, such as graph theoretical distance, valence, atomic number, and bond sum, computed via the open-babel toolkit.[72] The program then performs a minimization using the MP2/6-31G* method. From the optimized structure an iterative electrostatic potential fitting procedure using different levels of theory, MP2/6-311G** and MP2/aug-cc-pVTZh, and distributed multipole analysis is then performed to obtain accurate atomic multipoles.

The program performs a SMILES string database search to extract possible matching pre-existing covalent (e.g stiffness constant of bonds and angles) and vdW parameters. If no existing SMILES string matches are found, different procedures are used for covalent and vdW parameters.

In the case of covalent parameters, the equilibrium values for bonds and angles are taken from the initial QM optimized geometry. A minimization is then performed with those equilibrium values,and the output bond and angle values are compared to the input QM optimized values.

For vdW and torsion parameters, a fragmentation protocol is performed for large molecules. This fragmentation protocol is necessary due to the poor scaling of *ab-initio* models. This allows to obtain torsion, vdW, or tor-tor parameters from fragments of the parent and then transfer parameters from the fragments back to the parent molecule. The vdW fitting procedure involves the probing water molecules with the molecule of interest.

The torsion fitting is the last step as torsions are very sensitive to changes in the nearby chemical environment. A 1D and 2D torsion scan of the fragment are performed at ωB97X-D/6–311+G* and torsion parameters are optimized in order to match these *ab-initio* datas.

An overview of this parametrization scheme can be seen in Figure 1.2. While Poltype is using pre-existing AMOEBA parametrization, the development of new FF models often involves manual trial and error as well as semi-automatic fitting framework such as ForceBalance (FB).[73]

FB works by minimizing a cost function expressed as a sum of weighted mean-square errors over experimental and *ab-initio* data sets. However, this method requires additional manual tweaking of parameters, making FF parametrization a cumbersome process. To address this issue, there has been significant research focused on speeding up the parametrization procedure using machine learning techniques. Although some preliminary work in this direction has been done during this thesis, it will not be discussed here.

Recently, this parametrization procedure has been coupled with Nuclear Quantum Effects (NQEs) to develop a highly accurate water model, Q-AMOEBA. In the next Section we will give a brief review about NQEs and why they are important in the development of the next generation of FFs and especially machine learning-based ones.[74, 75, 76]

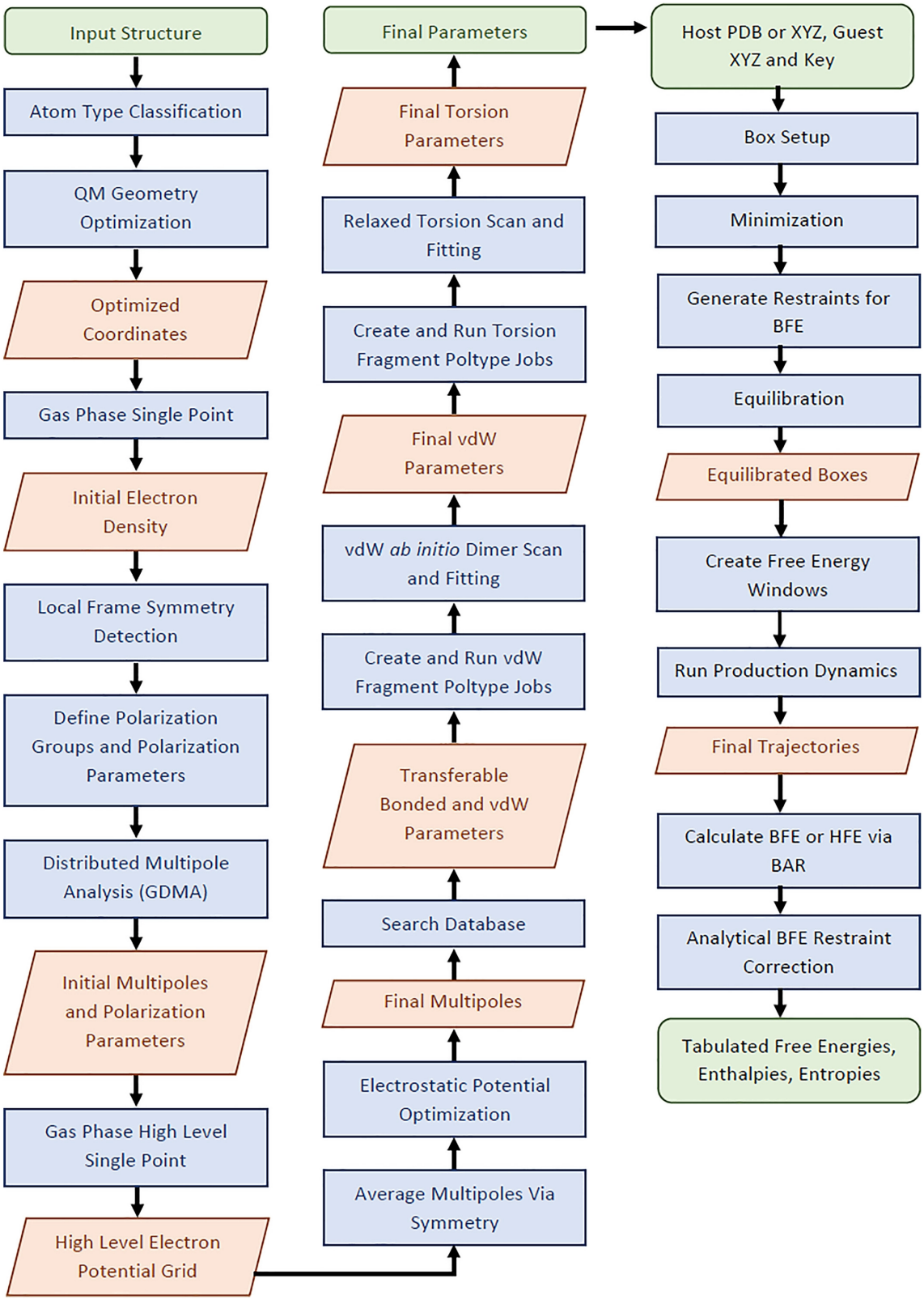


Fig. 1.2.: Overview of Poltype2 parametrization scheme including electrostatic potential fitting, and torsion scan procedure. Green boxes represent the input and output, red boxes indicate intermediate outputs, and blue boxes denote computations performed. Figure extracted from ref [71].

1.2.3 Nuclear Quantum Effects

One of MD's limitations is due to the use of Newton's equations of motion, which treats nuclei as a classical particules. The conceptual and computational complexity of the methods that account for Nuclear Quantum Effects (NQEs) explicitly has hindered their spread to a broad community. Reliable results can be obtained using Feynamn's Path-Integrals Molecular Dynamics framework (PIMD). PIMD provides a numerically exact reference for static properties, but its numerical cost can become very large compared to classical MD.
Thus, NQEs such as zero-point energy (ZPE) and tunneling are typically ignored in large scale simulations especially in biology where PIMD is too costly. However, neglecting these effects can have a significant impact on the computed properties. For instance, a chemical bond with a frequency of ω has a ZPE of $\frac{\hbar\omega}{2}$. In the case of the oxygen-hydrogen bond, with vibration mode of 3000 cm^{-1}, its ZPE is no more negligible compared to the ambient thermal energy k_BT. Thus neglecting the ZPE would be detrimental to the accuracy of the computations.

Path-Integral Molecular Dynamics

We will first explain the standard PIMD method mentioned earlier. This technique incorporates NQEs into the nuclei by representing each nucleus as a classical system of several fictitious particles connected by harmonic springs. This system is governed by an effective Hamiltonian derived from Feynman's path integral formalism. The quantum mechanical partition function in the canonical ensemble can be expressed as the partition function of a cyclic polymer composed of these fictitious particles. This cyclic chain of P classical particles exhibits an isomorphism with the quantum system, where each particle, also known as a bead, corresponds to a different imaginary time slice. The isomorphism is only exact when the number of beads, P, is infinite.[77, 78, 79, 80]
The path integral formalism samples the potential in an extended ring polymer phase space at a temperature $P \times T$. As such, P should be large enough to provide sufficient energy to account for the ZPE, an empirical criterion for convergence is that Pk_BT should be significantly greater than $\hbar\omega_0$, with ω_0 being the highest frequency of the system. Consequently, as temperature decreases, P needs to be relatively larger than $\frac{\hbar\omega_0}{k_BT}$. For systems with many degrees of freedom, the number of beads needed can be estimated using the highest vibrational frequency. One advantage of PIMD is that it provides numerically exact estimates for thermal equilibrium observables. However, its use is limited by the computational cost and scaling with the number of beads, making low-temperature and large-scale simulations computationally expensive. Additionally, dynamical properties are not directly accessible through the PIMD formalism, even though it is exact for static equilibrium properties.

The Quantum Thermal bath

In this section, we will discuss about a novel approach called Adaptive Quantum Thermal Bath (adQTB). Unlike the QTB, adQTB utilizes a criterion based on linear response theory to prevent the

leakage of zero-point energy. One major advantage of adQTB is its low computational cost which is essentially equal to that of a classical MD simulation.
In (ad)QTB simulations, the nuclei degrees of freedom follow a Langevin equation.

$$\mathbf{m}_i \frac{\partial^2 \mathbf{r_i}}{\partial t^2} = -\frac{\partial E}{\partial \mathbf{r_i}} - \mathbf{m}_i \gamma \frac{\partial \mathbf{r_i}}{\partial t}(t) + \mathbf{F}_i(t) \tag{1.41}$$

In this equation, E is the potential energy, the second term of the right part of the equation is a dissipative force (with friction coefficient γ) balanced by a random force $\mathbf{F}_i(t)$ that injects energy into the system. Here the random force has a correlation spectrum such as

$$C_{F_i,F_j} = m_i \gamma_i(\omega) \theta(\omega, T) \delta_{ij} \tag{1.42}$$

with $\gamma_i(\omega)$ a random force, δ_{ij} the Kronecker delta and $\theta(\omega, T)$ is the average thermal energy in a quantum harmonic oscillator at frequency ω and temperature T. The primary goal of the QTB is to consider ZPE contributions in a classical dynamics system by introducing an effective energy, $\theta(\omega, T)$, to each vibrational mode, rather than using the classical thermal energy, $k_B T$. However, the initial version of QTB was associated with some drawbacks that resulted in an inaccurate distribution of energy. This discrepancy can be quantified using linear response theory, where the deviation for each degree of freedom i is given by

$$\Delta_{\text{FDT},i}(\omega) = \text{Re}[C_{v_i F_i}(\omega)] - m_i \gamma_i(\omega) C_{v_i v_i}(\omega) \tag{1.43}$$

where $C_{v_i v_i}(\omega)$ represents the velocity autocorrelation function, and $C_{v_i F_i}(\omega)$ represents the cross-correlation spectrum with random force F_i. This deviation, $\Delta_{\text{FDT},i}(\omega)$, should be zero for all frequencies ω. However, this condition is not satisfied in QTB.
To address this issue, the adQTB method estimates the deviations $\Delta_{\text{FDT},i}(\omega)$ at regular time steps and adjusts the coefficients $\gamma_i(\omega)$ dynamically. Specifically, a negative deviation at frequency ω indicates an excess of energy, so the corresponding coefficient $\gamma_i(\omega)$ is reduced, and vice versa for positive deviations. This process is summarized in Figure 1.3 taken from.[81] The adQTB results are obtained once the $\gamma_i(\omega)$ are adapted such that the average deviation $\Delta_{\text{FDT},i}(\omega)$ vanishes.
The adQTB method has been successfully applied to various systems, such as water modeled by the q-TIP4P/F force field, as well as small molecules and proteins using reactive machine learning models.[74, 75, 82] In this thesis, we will utilize the adQTB method in the development of the Neural Network AMOEBA model, which will be discussed in Chapter 2.4.

As the complexity of systems being studied with MD simulations increases, it becomes increasingly important to balance accuracy with speed. While much effort has been devoted to improving accuracy through various physics-based models, such as polarization, parametrization strategies, and NQEs, parallelization and GPU-acceleration strategies are also essential to meet the growing computational demands. In the next section, we will delve into GPU-acceleration and high performance computing strategies, with a particular focus on their implementation within the Tinker-HP software package.

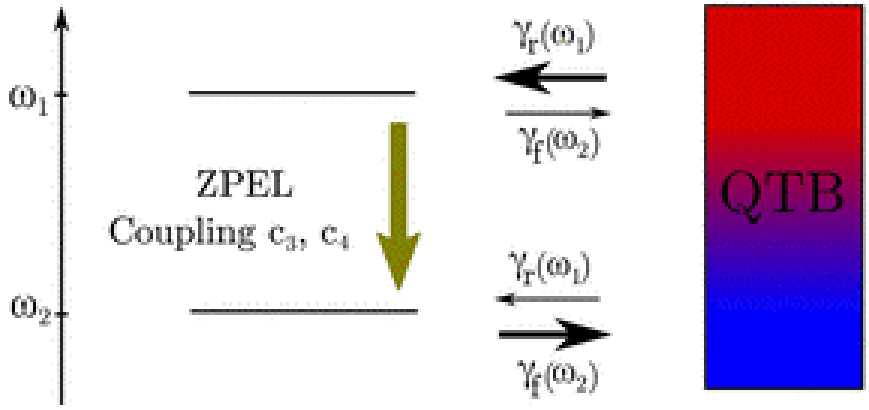


Fig. 1.3.: This figure illustrates the adQTB scheme for correcting the Zero Point Energy (ZPE) distribution. The $\gamma_r(\omega)$ values are adjusted to achieve the accurate energy distribution for each frequency. Figure extracted from ref [81].

1.2.4 GPU-acceleration and Parallelization Strategies

In order to simulate larger biological systems and to accelerate the evaluation of the potential, MD softwares such as Tinker-HP were extended toward high-performance computation. Parallelization strategies aim to efficiently distribute the workload between processor in order to fully take advantage of petascale and pre-exascale supercomputers. Tinker-HP is a massively MPI parallel code that can use thousands of CPU and have been extended more recently to the use of GPU. This extension will be discuss, briefly in this thesis in Chapter 3 Section A.

Domain Decomposition

To treat short-range interactions on large-scale parallel computers using distributed memory paradigm, several strategies have been developed. The Tinker-HP software uses a spatial domain decomposition method, where the simulation box is divided into 3D domains where each domain is assigned to a processor, which could be either CPU or GPU. The process then handles the calculation of forces and updates the coordinates of the atoms assigned to the domain at each time-step.
This approach is effective because it is based on the assumption of short-range interactions and on the fact that the atom positions do not change much between two consecutive time-steps. Thus, the forces on an atom are mainly originating from its nearest neighbors. If the cutoff for the short-range interactions, r_c, is greater than the size of a domain edge, which is often the case with a high number of processes, the communication volume scales like $\mathcal{O}(r_c^3)$ independently of the number of processes. Thus, the communication remains local, providing an advantage over other methods like atom distribution.
To perform a MD step using this method, we assume that the system is divided into 3D domains, with each processor assigned a block. In the first integration step, the local positions are updated, and the velocities are adjusted accordingly. This step may cause some atoms to change of block, requiring a local communication step through the reassignment to neighboring domains.

In the second step, forces are computed and used to update the velocities again. This step requires the process to know the positions of all atoms within the interaction cutoff r_c, which are communicated from processes assigned to domain that are within or equal to the cutoff distance. This step also involves local communications but may require more distant processes than the previous one. Once this is done, the algorithm loops back to the first step.

Midpoint method

Communications between processes can be a major bottleneck in the parallel calculations, resulting in a reduction in performance, especially when using a high numbers of processors. Tinker-HP uses the midpoint method, to reduce the communication volume in this last step.[83] This method selects the process responsible for computing the interaction between two atoms as the one assigned to the subdomain containing the center of the segment connecting the two atoms. Consequently, each process only needs to import information about atoms located at less than $\frac{r_c}{2}$ distance from its domain. This communication reduction is significant compared to the naive method, particularly when using a high number of processes. In addition, it gives good load balancing properties. The aim of load balancing is to distribute a set of computations over a set of processes thus minimizing the waiting time between communication as in biomolecular simulations each domain often contains the same number of atoms. Simulating large systems is usually limited due to memory requirement. In Tinker-HP, various strategies aim at managing this issue as the memory is partitioned among the processes and can be dynamically reallocated. For example, when computing non-bonded interactions, the neighbor lists, that are the most memory-intensive part of the program are reallocated as frequently as they are updated.
In the case of PFFs such as AMOEBA, another issue is coming from the array containing global parameters that is much bigger compare to non nolarizable FFs. Replicating these arrays for each processor would be inefficient for memory. Tinker-HP is using shared memory segments ensuring the array to be allocated only once per node and is thus accessible by every processor within the node, thereby drastically reducing memory requirements.

GPU-acceleration

In recent years a new paradigm has emerged to facilitate computation and programming on GPU devices. The parallel computing power of GPUs is constantly evolving and is almost doubling with each generation.
Indeed, the performance of microprocessors, e.g CPUs, is influenced by various factors such as their size, clock speed, core count, cache size, memory bandwidth, and instruction set architecture all contribute significantly to a CPU's performance. However, as CPUs' sizes continue to shrink, maintaining substantial performance gains with each new generation becomes increasingly challenging. This is due to several factors, including the difficulties in manufacturing and controlling ever-smaller features, the rise in power consumption as feature sizes decrease, and the limitations imposed by the

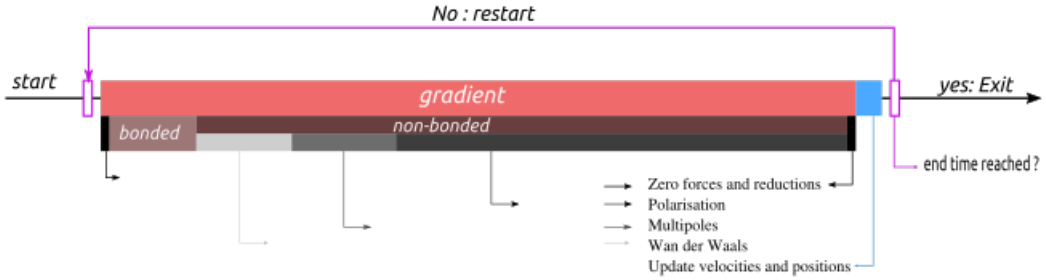


Fig. 1.4.: Breakdown of MD time-step contributions in Tinker-HP GPU simulations. Around 10% originating from bonded interactions and 90% from nonbonded interactions such as polarization, permanent multipoles electrostatics, and van der Waals. Among the nonbonded interactions, polarization is the most dominant component. Figure extracted from ref [12].

laws of physics. A critical component to improve is not the CPU itself but rather the speed and latency of memory. If a RAM memory design can match the speed of cache memory, CPUs would be able to unleash their full potential on almost every memory bounded applications. Unfortunately, memory technology and architecture significantly lag behind CPUs, creating a bottleneck that noticeably slows down overall CPU performance. Despite proposals for memory upgrades, none of them was able to reach the general public.

These limitations have prompted the development of GPUs (Graphics Processing Units) devices, aimed at addressing the shortcomings of CPUs. Nvidia's recent introduction of the GRACE chip represents a notable advancement in chip architecture. The GRACE chip integrates both the CPU and RAM onto a single large die. This innovative design holds the potential to effectively leverage the CPU's capabilities by extending RAM to a bandwidth of 900GB/s with a latency of 10 ns. By addressing memory limitations, we can potentially unlock the full potential of CPUs and achieve significant performance improvements. However, at present, GPUs have emerged as the primary focus of interest, particularly in scientific research.

Given the impressive computational power of GPUs compared to CPUs, the only way to fully leverage this power is to offload the entire computation to the device. Performing the computation on GPU without the need of communication with the CPU is called GPU-resident computations. Indeed, significant part of the computation workflow should not be performed on the CPU platform, as it would cause a bottleneck in performance and require multiple data transfers.

In most MD softwares, such as Tinker-HP, computing forces is the most time-consuming task. This is particularly true for the AMOEBA polarizable model, where force evaluation accounts for approximately 97% of a time step when executed sequentially on CPUs. Among these forces, nonbonded interactions, which include electrostatics (polarization and permanent multipoles) and van der Waals forces, comprise more than 90%, as illustrated in Figure 1.4. Polarization, in particular, is the dominant component of the nonbonded forces. As a result, significant efforts have been directed towards optimizing and porting non-bonded forces and polarization, as outlined in Chapter 2 and 3. Several studies have shown that many MD softwares can provide accurate results with reduced precision. Studies on polarizable force fields have only recently been conducted, one of which is presented in Chapter 3.

Mixed-precision (MP) mode is a technique that uses less accurate numbers, such as single-precision (SP) 32-bit representation, to calculate individual forces and energy contribution, while using higher precision, such as double-precision (DP) 64-bit, for accumulation.
Tinker-HP software initially provided a full DP mode that is used for highly-accurate reference simulations and a MP mode used for performing fast and long simulations. This MP mode is computing the individual forces and energies in SP while their sums, accumulation, is computed with DP.
Recently, another mode using fixed precision arithmetic (FPA) have been implemented. Indeed, GPUs were initially designed for image processing and gaming and did not require DP, leading to a focus on optimizing integer and SP calculations instead. To address this issue, Tinker-HP, along with other software such as AMBER and OpenMM, proposed the use of FPA, which replaces DP computations with integer-based ones for force component accumulation.[84] This approach fully utilizes the potential of newly wide audience GPUs, where floats are coded in integers. In contrast, MP is typically faster than FP on modern supercomputer GPUs, such as the V100, with native DP arithmetic precision. Thus, both FP and MP significantly improve performance on modern GPU hardware without sacrificing numerical accuracy, although their relative performance depends on the GPU used.
To leverage Tinker-HP to operate on multi-GPUs, direct communications between GPUs are made directly using a CUDA aware MPI implementation. In comparison with non nolarizable FFs, PFFs require more communication between processes due to the polarization solver computations.
However, when running on multiple nodes, inter-node communications become the bottleneck, as the interconnection between nodes is often much slower than that within a node. For example, AMOEBA's peak of performance is often hit when running on an entire node. Indeed, for example on the Jean-Zay supercomputer, each GPU comes with a 300 GB/s interconnection NVlink bandwidth while the interconnection among nodes is about 32 GB/s. Thus inter-node transit times is 10 times slower. Thus, in a multi-node context, the bottleneck clearly lies in the inter-node communications. This issue is the subject of an active research, and progress in compiler technology and the availability of large pre-exascale supercomputers may help to alleviate it in the future.
Finally, high performance computing and GPU computing have significantly accelerated the rise of machine learning, as models and database become increasingly large, sometimes with billions of parameters trained on even more data. An example is stochastic gradient descent algorithms which can use half-precision 16-bit floats, thus fully utilizing the potential of GPU computing.

1.3 General Machine Learning Methods and Tools

This section provides a brief overview of machine learning algorithms. It begins by presenting a definition of machine learning proposed by Mitchell [85] and referenced by Goodfellow et al.[86] We then explain the two main types of machine learning algorithms: unsupervised and supervised. The former was used in this thesis to extract information from data, such as clustering, and was used in combination with enhanced sampling techniques. The latter was used to develop machine learning

potentials. As machine learning is a broad subject, we only introduce the relevant methods used in this thesis.

1.3.1 Definitions

Machine learning can be defined as an algorithm that is able to learn from data. But, in his paper Computing Machinery and Intelligence, Alan Turing raises the question "Can machines think?". A widely accepted definition of learning in computer science was proposed by Mitchell in 1997 [85] and cited as reference in the famous book by Goodfellow et al.[86]: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E. The task T typically involves how the machine learning algorithm should process a collection of features, denoted by $\mathbf{x}$ which are measured or computed values associated with specific events. For example, in a regression task, the objective of the model is to predict a value given inputs. To accomplish this, the learning model aims to output a function f. The performance P, is a measure of how accurately the model can predict the correct output. Especially, in machine learning we are interested on the performance of the model on unseen data, which is evaluated using a separate dataset called the test set. This test set is distinct from the training set used to train the model. The distinction between unsupervised and supervised machine learning methods arises from the type of experience E that the algorithm is allowed to gain from the dataset during the learning process.

Unsupervised machine learning algorithms gain experience from dataset containing features but unlabeled. By unlabeled we mean that each data of the dataset has not been labeled by a specific property or measure. Such method aims to learn the intrinsic structure of the data for example by learning its probability distribution $p(\mathbf{x})$. Another popular type of unsupervised algorithms consist of partitioning the dataset into clusters which contains data similar to each other.

Supervised machine learning algorithms gain experience from dataset containing features but this time labeled. The label, or target, will be denoted by $\mathbf{y}$. In chemistry the features would be the Cartesian coordinates and the label would be, for example, the energy of the conformation. Some of the objectives of these algorithms are to either find the function f such as $\mathbf{y} = f(\mathbf{x})$ or estimate the joint distribution $p(\mathbf{x}, \mathbf{y})$.

However, it is important to note that the classification between supervised and unsupervised learning is in many cases not so clearly defined, as there is no objective criterion to determine whether a value is a target or a feature.

In the following, we will provide a brief overview of some of these methods, with a particular focus on the ones used in this thesis.

1.3.2 Unsupervised Algorithm

As explained previously, unsupervised learning algorithms aim at extracting information from a distribution of unlabeled data. Unsupervised learning is associated with tasks such as density

estimation, low dimensional representation, and clustering, where the focus is on exploring the underlying patterns and structure within the data. In this section we will present some low dimensional representation as well as clustering algorithms.

Principal Components Analysis

The Principal Component Analysis (PCA), is an unsupervised learning algorithm which find a low dimensional representation of the input data. This representation is created in a way where the elements are statistically independent and have no linear correlation with each other. To achieve complete independence, the algorithm must also eliminate nonlinear relationships between variables. In PCA, an orthogonal linear transformation is learned, projecting the input data point $\mathbf{x}$ onto a new representation Z. One of the fundamental properties of PCA is its ability to transform the data into a representation where the elements are uncorrelated. This characteristic is highly significant as it allows PCA to uncover the underlying factors of variation within the data by aligning the principal axes of variance with the basis of the new representation space associated with Z. Although correlation is an important form of dependency between data elements, it may not be sufficient to uncover more complex feature dependencies in representations. For this, we will need more than what can be done with a simple linear transformation. While PCA has been invented at the beginning of the 20th century, it is still widely used and found to be useful in many disciplines such as in enhanced sampling techniques. In the Chapter 3 we used it as a first guess for evaluating the Collective Variables (CVs) of the SARS-CoV-2 M^{pro}. It is also important to mention that the use of non-linear transformations for finding CVs is an active field of research [87, 88, 89]. One machine learning model that is commonly used in this framework is the Variational Autoencoder (VAE) [90].

DBSCAN

The Density-based spatial clustering of applications with noise (DBSCAN) is a widely used density-based clustering algorithm. It groups points in a given space based on their proximity. DBSCAN relies on two parameters, ϵ, which determines the maximum distance between points to be considered neighbors, and minPts, the minimum number of points required to form a cluster. The algorithm starts with an unvisited point and retrieves its ϵ-neighborhood. If the neighborhood contains a sufficient number of points (minPts), a cluster is created otherwise, the point is labeled as noise. It is important to note that a noise point may later become part of a cluster if it is found within the ϵ-neighborhood of another point. When a point is identified as part of a cluster, its ϵ-neighborhood is also assigned to that cluster. DBSCAN offers several advantages. Firstly, it does not require the number of clusters to be specified in advance. Secondly, it can discover clusters of arbitrary shapes and identify clusters within clusters. Additionally, DBSCAN is robust to outliers. Lastly, the algorithm only requires two parameters and is generally insensitive to the order of points in the dataset. DBSCAN along with its

variants HDBSCAN and OPTICS [91, 92] were used in the Chapter 3 to extract meaningful cluster, or macrostates, of the SARS-CoV-2 M^{pro}.

1.3.3 Supervised Algorithm

Overfitting, underfitting and regularization

One of the objective of Supervised learning algorithm is to find the function f which maps an input $\mathbf{x}$ (e.g molecular conformation) to a label $\mathbf{y}$ (e.g energies) through the optimization of some parameters θ: $\mathbf{y} = f(\theta; \mathbf{x})$. The parameters θ can be weights ω or biases $\mathbf{b}$, and usually both, explained in the next section, and are optimize to minimize what is called a loss function $\mathcal{L}$. One of the most well known method to minimize $\mathcal{L}$ is the gradient descent.

$$\omega \longleftarrow \omega - \epsilon \nabla_{\omega} \mathcal{L}(\omega; \mathbf{x}, \mathbf{y}) \tag{1.44}$$

where ϵ is the learning rate that is determining how large the parameters are updated. The performance of a machine learning model depends on two key factors: its ability to reduce the training error and to minimize the gap between the training and test error. These factors correspond to the central challenges in machine learning: underfitting and overfitting, show in Figure 1.5. Overfitting occurs when the gap between the test and training error is large while underfitting occurs when the training error is high. The concept of overfitting and underfitting is directly linked to the complexity of the model. Indeed, a model with higher complexity than its task will overfit while a model with low complexity with respect to its tasks will underfit. Controlling the complexity of a machine learning model aims at limiting the functions included in the hypothesis space. The different approaches that aims to control the complexity are known as regularization.

A major research effort has been concentrated on the development of regularization techniques. Most of regularization strategies limit the complexity of a model by adding norm penalty $\Omega(\theta)$ to the loss function.

$$\hat{\mathcal{L}}(\theta; \mathbf{x}, \mathbf{y}) = \mathcal{L}(\theta; \mathbf{x}, \mathbf{x}) + \alpha \Omega(\theta) \tag{1.45}$$

The most common strategy to limit this complexity is known as weight decay that consist of forcing the weights to be closer to zero by adding a L^2 penalty $\Omega(\theta) = \frac{1}{2}\|\omega\|_2^2$. The update of the weights then become.

$$\omega \longleftarrow (1 - \epsilon\alpha)\omega - \epsilon \nabla_{\omega} \mathcal{L}(\omega; \mathbf{x}, \mathbf{y}) \tag{1.46}$$

The weight decay term thus constraint at each step the weights by a constant factor that depends of the regularization parameter α.

Feed-Forward Neural Networks

Feed-Forward Neural Network (FFNN) are the most used models in the field of deep learning. The term feed-forward is because the model has a feedback from the outputs into itself. Networks stand

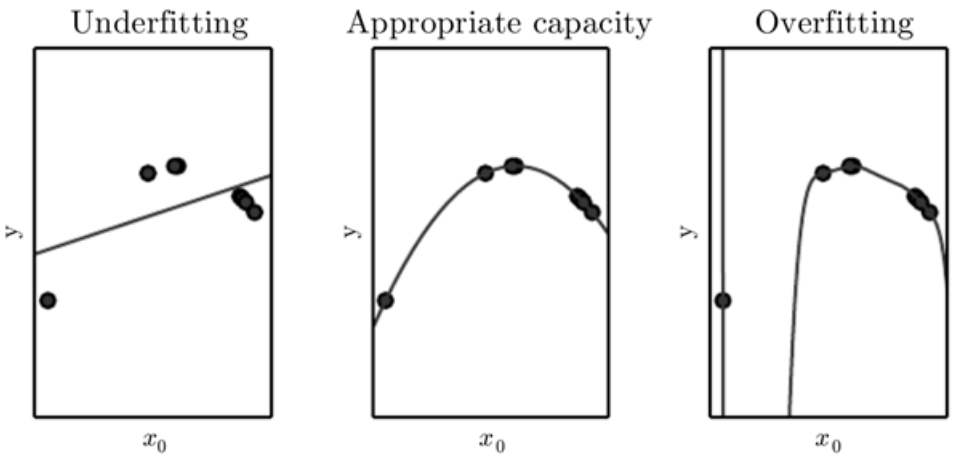


Fig. 1.5.: Three models were applied to a toy training dataset. This training dataset was generated by randomly sampling $\mathbf{x_0}$ values and determining $\mathbf{y}$ through a quadratic function. The left figure depicts underfitting, where a linear function was used to fit the data and thus fails to capture the curvature. The right figure exhibits overfitting, where a polynomial of degree 9 was used. The center figure demonstrates appropriate fitting, where a quadratic function was used. Figure extracted from ref [86].

from the fact that the model is a composition of functions, $f(\mathbf{x}) = f^{(n)} \circ f^{(n-1)} \dots f^{(1)}(\mathbf{x})$ where n is the number of layers. The last layer of the network is the output layer. The other layers are called hidden layers because they do not yield the target output. The training process objective is to drive the estimate $\hat{f}(\mathbf{x})$ to match the true function $f(\mathbf{x})$. Each layer of the FFNN is a vector to scalar function where each of its elements are called neuron as it resembles a neuron in the sense that it receives input from many other units and computes its own activation value. The choice of the functions $f^{(i)}(\mathbf{x})$ used to compute these representations is also coming from neuroscience. Most FFNN use an affine transformation controlled by learned parameters as $f^{(i)}(\mathbf{x})$, followed by a nonlinear function called an activation function $\mathbf{h} = \sigma(\mathbf{w}^\intercal \mathbf{x} + \mathbf{b})$ where $\mathbf{w}$ and $\mathbf{b}$ are the weights and biases of the linear transformation and σ is the activation function. Some popular activate function are the hyperbolic tangent, the sigmoid function, the softplus function, and Gaussians or the rectified linear unit (ReLU)[93]: $\sigma(x) = max\{0, x\}$

FFNNs provide numerous benefits especially for constructing machine learning potential. They exhibit great flexibility, offering a vast number of adjustable parameters. Moreover, the simple functional form of FFNN facilitates the computation of derivatives which are crucial in MD where the computation of forces are needed.

Message Passing Neural Network

Message Passing Neural Network (MPNN), a subclass of Graph Neural Network (GNN), is a class of model that use graphs as input data. These models have seen an increasing interest in the field of chemistry as molecule can be seen as a graph $G = (V, E)$ where the nodes V are the atoms and the edges E are the connection between the atom and all other neighboring atoms.

Each node $i \in V$ is represented by a hidden state $\mathbf{h}_i^t$ at layer t, and each edge $e \in V$ is represented by

an edge feature $\mathbf{e}_{ij}$. In the context of molecule $\mathbf{h}_i^t$ is the state of atom i and $\mathbf{e}_{ij}$ is often represented by the interatomic distance. The message passing neural network formalism is

$$\mathbf{m}_i^{t+1} = \sum_{i=0}^{N_V} M_t(\mathbf{h}_i^t, \mathbf{h}_j^t, \mathbf{e}_{ij}) \tag{1.47}$$

$$\mathbf{h}_i^{t+1} = U_t(\mathbf{h}_i^t, \mathbf{m}_i^{t+1}) \tag{1.48}$$

where M_t is a message function, U_t is a node update function and N_V is the number of neighboring nodes of the node i within a certain radius cutoff $r_{c,l}$ for large graph. As the messages are communicated over a sequence of t steps, the local receptive field of an atom node i, representing the set of effective neighbors to the atom's final state, increases approximately cubically with the effective cutoff radius, denoted as $r_{c,e}$. In a MPNN with N_{layer} message passing steps and a local cutoff radius of $r_{c,l}$, the resulting effective cutoff can be calculated as $r_{c,e} = N_{layer} \cdot r_{c,l}$. As a result, information from all atom nodes within this $r_{c,e}$ influences the state of the central atom node at the final layer of the network. This means that MPNNs have the capability to capture long-range interactions and many-body correlations, making them highly effective in extracting meaningful information from graphs. However, one major drawback is their cubic scaling with the number of nodes within the effective cutoff $r_{c,e}$, poses challenges in terms of memory consumption and scalability, particularly for large graphs.

1.4 Overview and Perspectives of Machine Learning Potentials

This section is dedicated to Machine Learning Potentials (MLPs), MLPs are expressions of the potential energy surface that provide both the potential energy and its analytic derivatives with respect to the atomic positions, through the use of machine learning algorithm. The parameters of the model are optimized using a dataset of reference electronic structure data, e.g ωB97X or CCSD(T)/CBS. In recent years, there has been a significant number of published MLP, based on various approaches such as high dimensional neural network potentials (HDNNPs),[20] Gaussian approximation potentials (GAP),[94] gradient-domain machine learning (GDML)[95] and SchNet.[96] This section introduces some of these MLP models and explains their general structure, which forms the basis of most of the work during this thesis.

1.4.1 General Structure of Machine Learning Potential

The first introduced MLP can be traced back to the seminal work by Doren et al. in 1995.[97] Their paper presented the first ML potential based on DFT for the adsorption of H_2 on a Si(100) surface.

This MLP made use of a single FFNN for the global PES.
In 2007, Behler and Parrinello [20]abandoned this use of a single FFNN and used the locality approximation, which states that the atomic interaction energies are mainly local. In this fashion the total energy of a system E_T can be written as a sum of local atomic contributions E_i predicted by a FFNN.

$$E_T = \sum_{i}^{N_{atoms}} E_i(G_i) \tag{1.49}$$

These local atomic contributions depend on the neighborhood of an atom within a certain cutoff radius R_c. This cutoff radius is a parameter that in principle has to be tested for each system. Although this local approximation looks crude, it turns out that for most systems cutoffs between 6 and 10 Å is sufficient to capture most of the energies.
Once the neighborhood of the atom is set, the atomic coordinates have to be converted to a suitable input for the FFNN which should obey the conditions of translational, rotational, and permutational invariance of the energy function. This encoding of the atomic environments is called an atomic environment descriptor or vector (AEV), denoted as G_i in equation 1.49.
Currently a major part of MLPs research focuses on the development of efficient and transferable AEV. These descriptors are often based on physically-driven hand-crafted functions but intensive work have been to directly learn them through machine learning such as MPNN. In the next Section we will present some key MLP models as well as their atomic descriptors.

1.4.2 Examples of Machine Learning Potentials

Behler-Parinello High Dimensional Neural Network Potential (HDNNP)

In addition to introducing the locality approximation, Behler and Parrinello made a significant contribution by introducing the atom-centered symmetry functions (ACSF) descriptor [98]. This descriptor enabled the construction of MLPs that follow the fundamental principles of translational, rotational, and permutational invariance of the energy function.
The combination of the locality approximation with ACSF descriptors allows for the utilization of separate FFNNs to express the energy contributions of individual atoms in a system. These atomic energies are then aggregated to obtain a short-range energy estimate.
In this scheme, the coordinates of each atom are transformed into a vector of symmetry function values, or AEV, denoted as G_i which depends on the coordinates of its neighboring atoms within the cutoff sphere and captures specific radial and angular chemical regions. Their functional form will be discussed in the next part. The resulting atomic symmetry function vector serves as the input for an atomic network, which shares the same architecture and weight parameters across all atoms of a specific chemical element. This ensures that atoms of the same element are treated as chemically equivalent and that their energy contribution is solely determined by their atomic environment.
When an atom is added to the system, the corresponding atomic neural network of the respective element is included or, equivalently, evaluated once for each atom of that element in the system.

Conversely, if an atom is removed, the associated atomic neural network is excluded. This flexible approach overcomes the limitations of earlier neural network potentials, which were typically designed for systems with a fixed number of atoms.
The development of HDNNP enables the application of models trained on small molecules to larger systems while offering a linear scalability with respect to the number of atoms.

ANI - ANI2x

Following the pioneering work of Behler and Parrinello, Smith et al. developed the ANI model, which uses a slightly modified version of the original Behler-Parrinello symmetry functions, represented as $G_i = G_1, ..., G_M$. The ANI model, being one of the first general MLP models which was found to give accuracy close to DFT while being transferable to various molecular systems.
In order to smoothly enforce locality, they first employed a differentiable smooth cutoff function.

$$f_c(\mathbf{r_{ij}}) = \begin{cases} 0 & ||\mathbf{r_{ij}}|| > R_c \\ \frac{1}{2}\cos\frac{\pi||\mathbf{r_{ij}}||}{R_c} + 0.5 & ||\mathbf{r_{ij}}|| \leq R_c \end{cases} \tag{1.50}$$

Here, $||\mathbf{r_{ij}}||$ denotes the distance between the central atom i and a neighboring atom j, and R_c represents the cutoff radius. Similar to HDNNP, the AEV is divided into radial and angular symmetry functions. The radial symmetry function employed here is the same as the one used in the original Behler-Parinello HDNNP.

$$G^{rad}_{i,m} = \sum_{j\neq i}^{N} e^{-\eta(||\mathbf{r_{ij}}||-\mathbf{r}_s)^2} f_c(\mathbf{r_{ij}}) \tag{1.51}$$

The parameter m corresponds to a set of parameters $/\eta, \mathbf{r}_s/$, where $\mathbf{r}_s$ denotes the shift of the Gaussian center relative to the central atom, and η represents the spatial extent of the Gaussian. To capture angular information more effectively, a slightly modified version of the HDNNP angular symmetry functions was employed.

$$G^{ang}_{i,m} = 2^{1-\xi} \sum_{j,k\neq i}^{N} (1 + \cos(\theta_{\mathbf{ijk}} - \theta_{\mathbf{s}}))^{\xi} e^{-\eta(\frac{\mathbf{R_{ij}}+\mathbf{R_{ik}}}{2} - R_s)^2} f_c(\mathbf{R_{ij}}) f_c(\mathbf{R_{ik}}) \tag{1.52}$$

$\theta_{\mathbf{ijk}}$ represents the angle between the central atom i and its neighbors j and k, while θ_s is used to center the maxima of the cosine function. The parameter ξ determines the width of the peak. ANI incorporates separate radial and angular symmetry functions for each atomic number, resulting in N_s radial and $\frac{N_s(N_s+1)}{2}$ angular sub-AEVs when considering N atom species. For further information about geometrical variable please refer to Figure 2 of ref [99].
The first ANI potential, ANI-1x, has been developed for simulating organic molecules containing H, C, N, and O elements [100, 101]. A recent extension, ANI-2x [102], has been trained to include three additional elements: S, F, and Cl.
The ANI potentials have demonstrated impressive accuracy in predicting a wide range of properties, often giving better performance than DFT. Although their architecture is based solely on FFNN,

one of the main factors contributing to their success is the dataset. The ANI dataset possesses a wealth of chemical information, encompassing fifty thousand different molecules and totaling tens of millions of conformations computed at the $\omega B97X$ DFT level, with a subset also calculated at the CCSD(T)/CBS level.
In Chapter 2, Section 2.2, when coupled with AMOEBA, ANI-2x exhibited accurate predictions of binding free energy for challenging host-guest systems, achieving chemical accuracy. The ANI dataset was also used to develop the Q-AMOEBA-NN model in Section 2.4.

DeepMD

Another notable MLP model, DeePMD, introduced by Weinan E. et al. [103, 104], has gained significant popularity in the condensed phase matter community. DeePMD has been optimized for large-scale simulations involving millions of atoms. One distinctive feature of DeePMD, compared to other MLP models, is that it avoids the use of hand-crafted symmetry functions to capture atomic environments [103, 104]. Instead, for a given atom j, its j neighbors within a specified cutoff radius are sorted based on their chemical species and inverse distances from the central atom. Subsequently, the central atom i is associated with a local frame (e_x, e_y, e_z), and the local coordinates of its neighbors are denoted as (x_{ij}, y_{ij}, z_{ij}). The local environment of atom i, denoted as $\{D_{ij}\}$, is then defined as:

$$\{D_{ij}\} = \left\{ \frac{1}{R_{ij}}, \frac{x_{ij}}{R_{ij}}, \frac{y_{ij}}{R_{ij}}, \frac{z_{ij}}{R_{ij}} \right\} \tag{1.53}$$

This set of $\{D_{ij}\}$ values serves as input to a FFNN to predict the atomic energy E_i.
One notable advancement of DeePMD is its high efficiency and scalability, achieved through the DeepMD-kit software. This software has demonstrated the capability to simulate systems involving tens of millions of atoms, such as water and copper, by using a highly optimized GPU code on the Summit supercomputer [105].
While the DeepMD-kit software is user-friendly and computationally efficient, it is black box, limiting the improvment by the community. Additionally, it is not natively integrated into existing MD software. In this thesis, DeepMD was integrated into Deep-HP, allowing for its coupling with biologically-relevant MD features present in Tinker-HP.

Equivariant Neural Networks

Equivariant Neural Networks primarily focus on the effects of invariance and equivariance concerning E(3), which denotes the group of rotations, reflections, and translations. Scalar quantities like potential energy are invariant to these symmetry operations, while vector quantities like atomic forces or dipoles are equivariant and transform accordingly when the atomic geometry changes. Mathematically, a function $f : X \to Y$ between vector spaces is called equivariant with respect to a group G if:

$$f(D_X[g]x) = D_Y[g]f(x) \quad \forall g \in G, \forall x \in X \tag{1.54}$$

where $D_X[g]x \in GL(X)$ represents the group element g representation in the vector space X. A function f is invariant if$D_Y[g]$ is the identity operator on Y, meaning the output remains unchanged when symmetry operations act on the input x. Most MLP ensure the invariance of predicted energies by only operating on invariant inputs. In the case of equivariant neural networks they can directly process non-invariant geometric inputs, such as displacement vectors, while respecting these symmetry as only employing E(3)-equivariant operations.
Different types of equivariant architectures exist such as PAiNN,[106] Spookynet,[107] NequIP[108] or Allegro.[109] For the sake of simplicity, in the following we will only discuss the architecture of NequIP and Allegro. In these models, each atom is associated with feature vectors composed of tensors of various orders, including scalars, vectors, and higher-order tensors. The feature vectors take the form of a direct sum of irreducible representations, or irreps, of O(3). The irreps of O(3) and their associated features are indexed by a rotation order $l \geq 0$ and a parity $p \in (1, 1)$.
A crucial operation in equivariant networks is the tensor product of representations.

This operation combines two tensors $\mathbf{x}$ and $\mathbf{y}$, each with irreps l_1, p_1 and l_2, p_2, respectively, to produce an output tensor inhabiting an irrep l_{out}, p_{out} that satisfy some special condition. In a mathematical sense, this operation is computed through contraction with the Clebsch-Gordan coefficients:

$$(\mathbf{x} \otimes \mathbf{y})_{l_{out},p_{out}} = \sum_{m_1,m_2} \begin{pmatrix} l_1 & l_2 & l_{out} \\ m_1 & m_2 & m_{out} \end{pmatrix} \mathbf{x}_{l_1,p_1} \mathbf{y}_{l_2,p_2} \tag{1.55}$$

This tensor product has two key properties, it is bilinear (i.e linear in both $\mathbf{x}$ and $\mathbf{y}$), and it combines tensors inhabiting different irreps in a symmetrically valid manner. Several simple operations can be expressed using the tensor product, such as

- scalar-scalar multiplication: $(l_1 = 0, p_1 = 1), (l_2 = 0, p_2 = 1) \longrightarrow (l_{out} = 0, p_{out} = 1)$
- vector dot product: $(l_1 = 1, p_1 = -1), (l_2 = 1, p_2 = -1) \longrightarrow (l_{out} = 0, p_{out} = 1)$
- vector cross product: $(l_1 = 1, p_1 = -1), (l_2 = 1, p_2 = -1) \longrightarrow (l_{out} = 1, p_{out} = 1)$

For instance, the message function $M_t(\mathbf{h}_i^t, \mathbf{h}_j^t, \mathbf{e}_{ij})$ in the NequIP model uses the tensor product to define a message from atom j to atom i as the tensor product of equivariant features of the edge $\mathbf{e}_{ij}$ and the equivariant features of the neighboring node j.

Force-Field-Enhanced Neural Network InteraXions

Finally, we will present the Force-field-Enhanced Neural Network Interactions (FENNIX)[82] which is a model recently developed by our group. It employs a local multi-output equivariant neural network to predict local pairwise energy contributions, charges and atomic volumes. To improve its predictive capabilities and transferability, the model incorporates physical models by integrating electrostatic and dispersion energy terms. FENNIX is a modified version of the Allegro equivariant model, which serves as general embedding of atomic pairs. The embedding is subsequently input into separate neural networks that predict the target properties.

A notable enhancement of the FENNIX model lies in its inclusion of NQEs through the adQTB method. This addition has found to be crucial for accurately calculating thermodynamic properties since the model solely relies on *ab-initio* data. In addition, FENNIX introduces a positional encoding scheme that encodes coordinates in the periodic table through cosine and sine functions. The idea behind it is that assigning similar encodings to species that share a row or column can aid in generalization and facilitate the transfer of knowledge from one species to another. As a result, this approach may reduce the amount of training data needed.
The model is also combined with physical charge penetration, dispersion and electrostatic models thus enabling its transferability to a broader class of system and allow to capture at the same time both short range and long range interactions. The model was able to accurately generalize to condensed phase simulation, solvated organic molecules and solvated proteins.

1.4.3 Chemical Databases

As showed earlier, MLPs have proven to be highly efficient, surpassing the accuracy of long-standing physical models. However, it is important to note that the performance of MLPs is heavily dependent on the quality and quantity of data used during training stage. In the following sections, we will present significant datasets that were used and expanded upon during this thesis.

The ANI Datasets suite

ANI-1 dataset

The original ANI-1 dataset [100] is constructed through an exhaustive sampling process of a subset of the GDB-11 database, focusing on molecules with 1 to 8 heavy atoms comprising the atomic species C, N, and O. This selection results in a subset of 57,947 starting molecules, all of which are in a neutral state. The molecular structures undergo pre-optimization using non nolarizable Force fields (FF). Subsequently, geometry optimization is carried out at the ωB97x/6–31 G(d) level of theory, resulting in a total of 57,462 structurally optimized molecules. To generate a diverse set of conformers for each molecule, normal mode sampling computations are performed. Overall, the ANI-1 dataset encompasses more than 20 million conformations. In addition of being the building block of the ANI-1 potential, the availability of this dataset, which can be easily downloaded, contributes to its popularity. Consequently, it has been widely adopted in numerous MLP applications.
However, the ANI-1 dataset does have limitations. Firstly, the ωB97x/6–31 G(d) model lacks accuracy as it does not consider dispersion interactions and employs a very small basis set which would omit critical energy components. Additionally, while starting from the GDB database is a suitable approach for developing MLPs focused on small molecules, it lacks certain chemical insights that may be crucial for constructing a general-purpose MLP, especially to model water or biological systems.

ANI-1ccx and ANI-1x datasets

To address the initial limitations of the ANI-1 dataset, Smith et al. introduced two extensions, the ANI-1x and ANI-1ccx datasets [2]. The ANI-1x dataset encompasses five million molecular conformations computed using the ωB97x/def2-TZVPP method, which employs a larger and more accurate basis set compared to ANI-1. These conformations were generated through an active learning algorithm, where the ANI potential is iteratively trained to determine which new data should be included in future versions of the training dataset. The active learning algorithm incorporates four sampling techniques: high-temperature molecular dynamics simulations, normal mode sampling, dimer sampling combined with molecular dynamics (which considers intermolecular interactions by placing two randomly selected molecules from the GDB database in close proximity), and torsion sampling (incrementing the angle by 10 degrees across the entire torsion profile). This active learning process enhances the diversity of molecular conformations in the ANI-1x dataset compared to the original ANI-1 dataset.
On the other hand, the ANI-1ccx dataset is a carefully selected 10% subsample of the ANI-1x dataset, recomputed using the highly accurate CCSD(T)/CBS level of theory. Furthermore, in addition to using a higher level of theory, both dataset provide a wider range of molecular properties beyond energy, including forces, atomic volumes, and charges. This significantly increases the amount of available data, making it in practice larger than the ANI-1 dataset. Notably, it is known that forces are even more crucial than energy for training general-purpose MLP models, making this dataset particularly well-suited for molecular dynamics simulations. The ANI-1ccx dataset was also used to construct several models developed in this thesis, such as the DNN-MBD model discussed in Chapter 2 Section 2.1.

SPICE Dataset

The ANI dataset has proven to be valuable for constructing MLP models for small molecules. However, for drug design and biomolecular applications it is not sufficient to capture all meaningful biochemical insights. In this context, a careful and specialized dataset is required, encompassing amino acids, dipeptides, water probing, and solvated ions. To address this need, the SPICE dataset [110] was introduced recently, aiming to capture the energetics of molecular environments relevant to small molecules interacting with proteins.
The SPICE dataset consists of multiple subsets, each designed to provide specific information. One subset focuses on dipeptides, aiming to cover a wide range of covalent interactions commonly found in proteins. It includes all possible dipeptides formed by the 20 natural amino acids and their common protonation variants.
Another subset within the SPICE dataset focuses on solvated amino acids, specifically sampling protein-water and water-water interactions. These non-covalent interactions play a critical role in protein simulations, necessitating thorough sampling. This subset includes each of the 26 amino acid variants solvated with 20 water molecules.
The PubChem subset comprises a diverse collection of small, drug-like molecules extracted from the

PubChem database. These molecules are filtered based on specific criteria, such as having a size no larger than 50 atoms and consisting only of the elements Br, C, Cl, F, H, I, N, O, P, and S.
Additionally, the SPICE dataset incorporates a subset from the DES370K dataset [111], which includes dimers and monomers, providing extensive sampling of non-covalent interactions among diverse chemical groups. These subsets complement the dipeptides and PubChem molecules, primarily providing information about covalent interactions. The dataset also includes distance scans of ion pairs.
Finally, the SPICE dataset uses a very high level of *ab-initio* close to CCSD(T)/CBS as forces and energies were computed with ωB97M-D3(BJ)/def2-TZVPPD. Alongside forces and energies, the dataset includes additional properties for each conformation, such as MBIS charges, atomic dipoles, quadrupoles, and octopoles.
The SPICE dataset serves as a preferred choice for developing biomolecular MLP models and has been employed in the development of the Q-AMOEBA-NN model presented in Section 2.4.

To conclude this chapter, MLPs have matured significantly, and their accuracy can match state-of-the-art electronic structure models while providing significant speed improvements. This progress has been facilitated by the availability of large, diverse, and more accurate datasets, as well as the development of novel and efficient neural network architectures like equivariant neural networks that have expanded their capabilities. Furthermore, there has been a growing global investment from companies and governments in various aspects of the field, including AI-specialized GPU chips, model compressibility, and model architecture, which will further push the boundaries of MLPs in the near future.
However, it is important to note that no matter how accurate and fast the next generation of MLP models becomes, they may never match the speed of current non nolarizable FF models. Therefore, to access thermodynamic properties, enhanced sampling techniques will always be necessary.

1.5 Enhanced Sampling Methods

As explained before, no matter how accurate a model may be, the sampling of the conformational space of a molecular system is equally crucial. This is because, firstly, some high-energy barriers may be impossible to overcome through conventional MD simulations alone. Additionally, a thorough sampling of the conformational space is necessary to access thermodynamic properties and thus comparing simulation with experimental data. Enhanced sampling algorithms are the techniques that aim to improve the sampling efficiency of MD simulations. In the following sections, we will introduce fundamental principles of enhanced sampling techniques, provide an overview of current methods, and explore the potential of machine learning in enhancing their capabilities.

1.5.1 Free Energy and macrostates

Enhanced sampling techniques can be roughly divided into two categories.[112, 50] The first category comprises exploratory schemes that aim to discover unexplored regions of the conformational space. The second category involves schemes that enable the estimation of probability distributions and by extent free energies. To delve into the detail of these schemes it is important to introduce relevant notions and notations.

Macrostates can be described as experimentally distinguishable states of a molecule of its conformational space. A macrostate is a collection of microstates and is associated to macroscopic thermodynamic variables.
The Helmholtz free energy F is a thermodynamic property of a macrostate of a system in the canonical ensemble NVT. The Gibbs free energy G is the equivalent quantity in the isothermal–isobaric ensemble NPT. It represents its statistical weight compared to the other macrostates.

$$F = -\frac{1}{\beta}\int_{\Sigma} e^{-\beta E(\mathbf{r})} d\mathbf{r} \tag{1.56}$$

Where Σ is the subset of the configuration space that encompass the macrostate. As F is only defined up to a constant, the only physically relevant quantity is the free energy difference between two macrostates A and B.

$$\Delta F = F_A - F_B = \frac{1}{\beta}\ln\left(\frac{\int_{\Sigma_A} e^{-\beta U(\mathbf{r})} d\mathbf{r}}{\int_{\Sigma_B} e^{-\beta U(\mathbf{r})} d\mathbf{r}}\right) \tag{1.57}$$

The free energy difference is a key quantity in molecular dynamics as it can help to evaluate a model performance. Indeed, free energies are experimental observable thus simulations can be directly compared with "true" data. In certain cases, the system is biased towards specific regions of the conformational space, making the identification of macrostates relatively easy. An example of this is when using umbrella sampling, which will be explained later. However, in a broader context, characterizing these macrostates can be much more complex. One way to identifying these macrostates is by projecting the simulation into a low-dimensional space, known as collective variables. Through visual inspection or by clustering algorithms, these macrostates can be extracted from the collective variables. In the next section, we will define collective variables and discuss various methods for their construction.

1.5.2 Collective Variables

Extracting physically meaning full macrostates from molecular dynamics simulations is not an easy task due to the high-dimensional nature of the configuration space $3N$, with N the number of atoms. However, their long-time evolution can be understood trough some slow collective modes that arise from cooperative couplings between group of atoms. Thus, the understanding of a system can be

limited to a low n-dimensional manifold, where $n \ll 3N$, called collective variables (CV). This dimensionality reduction is done by projecting $\mathbf{r}$ on a set of $\xi = (\xi_1, \xi_2, \dots \xi_n)$ such as:

$$\mathbf{s} = \xi(\mathbf{r}) \tag{1.58}$$

CVs are usually defined using handcrafted functions based on chemical intuition, such as angles, distances, or Root-Mean-Square Deviation (RMSD) of the backbone of a molecule. Another approach involves combining these initial CVs through linear or nonlinear combinations to map them into a lower-dimensional space, allowing for a more concise representation.
While these approaches are generally effective, they may not fully capture complex chemical phenomena, especially in the field of biochemistry. We will provide in Section 1.5.10 an overview of data-driven techniques that employ machine learning algorithms for the discovery of CVs.

1.5.3 Estimating Free Energy Differences

The estimation of the difference in free energy between two macrostates forms the fundamental basis of computational chemistry. In fact, many chemical quantities are directly linked to this free energy difference. The probability of finding a system in one state or another is dictated by the free energy between those states. As a result, free energy differences are closely related with various chemical properties like solubility, adsorption coefficient, and binding constants. In this part, we will introduce estimators for calculating free energy differences using ensembles generated by MD. These estimators depend on the assumption of overlap, which means that configurations have a substantial probability of existing in both of these regions of the conformational space. In the subsequent explanation, we will represent the two states as i and j, characterized by the reduced energies u_i and u_j respectively, with N_i and N_j number of samples, $u_i(\mathbf{r}) = \beta_i E_i(\mathbf{r})$ and $\Delta u_{ij} = u_i - u_j$. The overall energy difference between these states is defined as follows:

$$\Delta F_{ij} = -\ln\left(\frac{\int e^{-u_j(\mathbf{r})} d\mathbf{r}}{\int e^{-u_i(\mathbf{r})} d\mathbf{r}}\right) \tag{1.59}$$

Free energy Perturbation

In free energy perturbation, often called exponential averaging, the previously defined free energy difference can be written as an ensemble average over the state i.

$$\Delta F_{ij} = -\ln\left(\frac{\int e^{-u_i(\mathbf{r})} e^{-(u_j(\mathbf{r}) - u_i(\mathbf{r}))} d\mathbf{r}}{\int e^{-u_i(\mathbf{r})} d\mathbf{r}}\right) \tag{1.60}$$

$$= -\ln\langle \Delta u_{ij}(\mathbf{r}) \rangle_i \tag{1.61}$$

This state average can be estimated numerically through a formally exact expression, in the limit $N_i \to +\infty$

$$\Delta F_{ij} = -\ln \frac{1}{N_i} \sum_{n=1}^{N_i} e^{-\Delta u_{ij}(\mathbf{r}_n)} \tag{1.62}$$

While this approach provides a mathematically exact solution and easy to understand, it is also one of the least efficient methods in practice. Its convergence relies on the distribution of $\Delta u_{ij}(\mathbf{r})$, on the degree of overlap between states and thus converge poorly with the number of samples. Unless the potential energy of all the conformations are within $2k_BT$, the exponential average should be avoided. To alleviate this issue, instead of computing the ensemble average from a single state, the Bennett's Acceptance Ratio (BAR) estimator [113] compute the ensemble average of both states $\Delta u_{ij}(\mathbf{r}_n)$ and $\Delta u_{ji}(\mathbf{r}_n)$.

Bennett's Acceptance Ratio (BAR)

BAR works under the principal that there is a pathway connecting the two reduced potentials u_i, u_j of a given conformation. BAR is the estimator with the lowest variance estimator to compute free energy difference between two states. This is done by finding $c = \Delta F_{ij}$ self-consistently through the equation

$$\sum_{i=1}^{N_j} \frac{1}{1 + e^{-(\Delta u_{ij} + c)}} - \sum_{j=1}^{N_i} \frac{1}{1 + e^{-(\Delta u_{ij} - c)}} = 0 \tag{1.63}$$

Finding c can be done in many ways available in different codes.[114, 115] From both a practical and theoretical perspective, the BAR method is a superior estimator compared to the exponential average. It converges faster, is less noisy, and requires a smaller amount of phase space overlap.

Multistate Bennett Acceptance Ratio (MBAR)

The Multistate Bennett Acceptance Ratio (MBAR)[116] is an extension of BAR to multiple states and offers a solution for computing the free energies F_i of all states, even unsampled ones, simultaneously.

$$F_i = -\ln \left(\sum_{n=1}^{N} \frac{e^{u_i(x_n)}}{\sum_{k=1}^{M} N_k e^{F_k - u_k(x_n)}} \right) \tag{1.64}$$

Here N represents the number of configurations at any of the M states. Since there is one equation for each free energy F_i of each states, this results to a series of M equation that need to be solved to determine the ensemble of F_i.
MBAR exhibits the lowest variance among the previously mentioned methods. Additionally, one of the significant advantages of MBAR is its capability to compute uncertainties of ΔF_{ij} by taking

into account the correlations between F_i and F_j due to their simultaneous estimation. A detailed comparison of these free energy estimators is presented in Chapter 3, Section 3.3 for the study of the alanine dimer and the CD2-CD58 protein system. After describing free energy estimators, an open question arises, what are the techniques to enhance the exploration of the conformational space?

1.5.4 Out-of-equilibrium Sampling Methods

Out-of-equilibrium enhanced sampling methods constraint a system to follow a given CVs or alchemical parameters.[117] These methods modify the original distribution and do not converge to an equilibrium ensemble. The behavior of out-of-equilibrium methods depend on the rate of the transformation. Outside the two limit, infinitely slow or fast switching, the free energy difference can be estimated by assigning weights to non-equilibrium trajectories. One well-known example is the alchemical free energy method used for computing binding and solvation free energies, examples can be found in Chapter 2 and Sections 2.2 and 2.3. The idea is to choose a thermodynamical pathway by introducing a thermodynamic parameter, λ, which smoothly connects two states 0 and 1 through a -dependent potential $\mathbf{E}(\mathbf{r}, \lambda)$ such that $\mathbf{E}(\mathbf{r}, 0) = \mathbf{E}_0(\mathbf{r})$ and $\mathbf{E}(\mathbf{r}, 1) = \mathbf{E}_1(\mathbf{r})$. In the case of computing solvation free energy of a ligand in water, state 0 is usually the ligand in water and state 1 is the ligand alone in vaccum. In the Free Energy Perturbation formulation, the transformation can be divided into a series of M steps, each associated with a specific $\lambda = \{\lambda_1, \lambda_2, ..., \lambda_M\}$, ranging from 0 to 1. These steps ensure that there is enough overlap in phase space between neighboring intermediate λ states. To achieve this, separate simulations are performed for each λ-window corresponding to a particular λ value, utilizing forces derived from the potential energy $\mathbf{E}(\mathbf{r}, \lambda)$. One straightforward approach is to use a linear interpolation between these two end states. However, it is well established that the linear alchemical transformation pathway presents practical issues that can be mitigated by incorporating the so called softcore potentials for non-bonded interactions. Numerous softcore potential forms have been proposed to soften these interactions.[118] In Tinker-HP and for AMOEBA, this is done by doing a y double-decoupling.[119] First, the electrostatic interactions between the ligand and its environment (e.g water) are decoupled by scaling the electrostatic parameters, e.g charges and multipoles. Subsequently, the vdW interactions between the ligand and its environment are scaled using a softcore potential. For hydration free energy computations, the intra-ligand electrostatic contributions are reintroduced by gradually increasing the electrostatic parameters for the ligand alone in the gas phase. Following these simulations, the BAR or MBAR methods, as discussed in the previous section, are employed to calculate the free energy difference between each adjacent state.

1.5.5 Non-adaptive Biasing Potential Methods

Non-adaptive biasing potential methods are designed to flatten the energy landscape by modifying the original potential. Sampling is biased by introducing an external bias potential, which requires careful unbiasing schemes. One of the pioneering non-adaptive biasing methods is the Umbrella Sampling method, which was introduced by Torrie and Valleau in 1977 [120]. Accelerated MD and

Gaussian-accelerated MD methods are currently among the most widely used approaches in this category of sampling strategies. These methods modify the potential energy through user-defined parameters, to effectively lower energy barriers either for specific transitions or within a specified energy range. By carefully applying reweighting techniques, unbiased statistics can be obtained. In the following section, we will provide a brief overview of both methods, which were incorporated into Tinker-HP during this thesis and modified to facilitate their integration with polarizable force fields and multi-time-step integrators, as described in Chapter 3, Section 3.3.

Accelerated Molecular Dynamics

Accelerated MD (aMD)[121] is an enhanced sampling technique that often involves adding an external boost potential to smooth the potential energy surface. The boost potential reduces energy barriers and facilitates transitions between different low-energy states. This enables aMD to sample distinct biomolecular conformations and rare barrier-crossing events that are difficult to access through conventional MD. In its original form, aMD applies a non-negative boost potential, denoted as $\Delta U^{aMD}(U(\mathbf{r}))$, to the potential energy surface if the system's potential energy is below a threshold energy E. The modified potential, denoted as $\tilde{U}(\mathbf{r})$, is defined as

$$U'(\mathbf{r}) = U(\mathbf{r}) + \Delta U^{aMD}(U(\mathbf{r})) \tag{1.65}$$

with $\Delta U^{aMD}(U(\mathbf{r}))$ is the boost potential

$$\Delta U^{aMD}(U(\mathbf{r})) = \begin{cases} 0 & U(\mathbf{r}) \geq E \\ \frac{1}{2}\frac{(E-U(\mathbf{r}))^2}{\alpha+E-U(\mathbf{r})} & U(\mathbf{r}) < E \end{cases} \tag{1.66}$$

The threshold energy E determines the affected portion of the potential surface, while the acceleration factor α determines the shape of the modified potential. It is important to note that α cannot be set to zero to avoid a discontinuous derivative of the modified potential. Several flavors of aMD have been developed, such as aMDd (boosting on the dihedral potential), aMDT (boosting on the total potential), and aMDdual (simultaneous boosting on both the dihedral and total potentials) [122].
Although aMD has shown significant improvements in conformational sampling, it suffers from large energetic noise during the reweighting process. The boost potential applied in aMD simulations typically ranges from tens to hundreds of kcal/mol, which is significantly higher than the one used in other biasing simulation methods involving CVs, which typically range in the order of several kcal/mol. Accurately reweighting aMD simulations, especially for large proteins, is a challenging task.

Gaussian-accelerated Molecular Dynamics

To address the limitations of aMD, Miao et al. introduced the Gaussian-accelerated Molecular Dynamics (GaMD).[123] GaMD smooths the potential energy surface by incorporating an external

harmonic boost potential. The key idea behind GaMD is to employ a boost potential that follows a near-Gaussian distribution, enabling the use of an efficient reweighting strategy based on the second-order cumulant expansion. In GaMD, the modified potential takes the following form,

$$U'(\mathbf{r}) = U(\mathbf{r}) + \Delta U^{GaMD}(U(\mathbf{r})) \tag{1.67}$$

with $\Delta U^{GaMD}(U(\mathbf{r}))$ the boost potential defined as

$$\Delta U^{GaMD}(U(\mathbf{r})) = \begin{cases} 0 & U(\mathbf{r}) \geq E \\ \frac{1}{2}k(E - U(\mathbf{r}))^2 & U(\mathbf{r}) < E \end{cases} \tag{1.68}$$

The two adjustable parameters in GaMD, namely k and E, are determined following a specific procedure [124]. To control the boost intensity, a user-specified upper limit denoted as σ_0, which is typically set to a predefined value like $10k_BT$, prior to the simulation. It is important to ensure accurate reweighting using the cumulant expansion by satisfying the condition $\sigma_{\Delta V} < \sigma_0$, where $\sigma_{\Delta V}$ represents the standard deviation of ΔU^{GaMD} [124, 125, 126]. Similar to aMD, GaMD offers various modes [127, 128]. More recently, a new mode called LiGaMD has been introduced, which applies the boost specifically to ligand non-bonded interactions [129].
The general framework of GaMD makes it well-suited for the development of hybrid schemes and variants, lengthly explain in the next Sections, such as replica-exchange umbrella sampling GaMD (GaREUS) [130], Ligand GaMD (LiGaMD) [129], and Peptide GaMD (Pep-GaMD) [131].
A previous challenge encountered with GaMD was its inability to be coupled with multi-time step integrators. This issue is particularly problematic when coupling GaMD with PFF as multi-time step integrators play a crucial role in accelerating simulations. In Chapter 3, Section 3.3, a new GaMD mode is introduced specifically designed to address this concern. This mode aims to seamlessly integrate with multi-time step integrators and, more broadly, with PFFs.

1.5.6 Adaptive Biasing Potential

The following section will provide an overview of adaptive biasing potential methods. While these methods were not used during this thesis, it is important to mention some of them as they play a crucial role in molecular dynamics. Adaptive bias simulations also involve biased sampling. However, unlike the previously mentioned methods, here, the bias is not predetermined or fixed but is instead learned and adjusted in real-time during the course of the simulation. This adaptive nature allows the bias to be continuously optimized based on the evolving behavior of the system, resulting in more efficient and effective sampling strategies.

Metadynamics

Metadynamics is part of adaptive biasing potential methods and is one of the most widely used enhanced sampling technique with numerous variants developed over the years.[132, 133, 134] In

metadynamics, the sampling is enhanced along a few selected CVs. This is achieved by introducing a time-dependent external bias potential that counteracts the free energy surface. The bias potential is constructed as a sum of repulsive Gaussian kernels, which are periodically added at the current position in the CV space. The bias potential provides direct estimates of the free energy surface as a function of the chosen CVs. Additionally, reweighting techniques can be employed to obtain the free energy surface for any other set of CVs. In certain cases, simulation time can be rescaled to extract rare-event kinetics from biased metadynamics simulations.

1.5.7 Replica Exchange

Another widely known sampling techniques are the replica exchange methods, which preserves the original configurational distribution while enhancing sampling by exploiting transitions to other ensembles. In replica exchange simulations, a total of K MD simulations are performed in parallel, called replicas, with each simulation representing a distinct thermodynamic state. Through the application of the Metropolis criterion, configurations are exchanged between neighboring thermodynamic states. In some cases, data collected from all states is crucial for calculating observables, while in other cases, only one simulation focuses on sampling a specific state of interest, while the remaining simulations are employed solely to assist the sampling process.

In standard replica exchange simulations, the exchanges are exclusively process between currently neighboring states. For instance, a common approach involves exchanges between the sets of state index pairs $(k, k+1)$ or $(k-1, k)$ with equal probability. Each state exchange attempt is carried out independently, and the acceptance of the exchange between states k and $k+1$ (or $k-1$) associated with configurations $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, respectively, follows the Metropolis criterion.

A major limitation of replica exchange is the requirement for simulations to possess some degree of overlap with each other in order for exchanges to occur with reasonable probabilities. Insufficient overlap results in an inefficient exchange scheme, where the probability of acceptance between certain pairs of replicas approaches zero. Proper selection of the replica spacing in the auxiliary variable is essential to avoid scenarios where many replicas remain in a few states throughout the entire simulation, indicating poor global overlap. A necessary check for global overlap involves ensuring that each individual simulation can transition between different states, ideally multiple times within the same simulation.

Additionally, it is important to note that in the development of many replica exchange methods, there is often an assumption of a natural ordering of states. This assumption allows for the identification of configurations resulting from simulations at state k as being more similar to configurations generated in state $k\check{}1$ and $k+1$ than to any other states. While this ordering is straightforward when states are defined along a single alchemical variable (λ) or temperature (T), it may not exist in more complex cases where thermodynamic states are defined in a multidimensional CV space. In such cases, certain schemes for exchange in replica exchange may not be applicable.

Replica exchange is widely employed as one of the most common expanded ensemble methods and

is implemented in various MD package. In a similar way to replica exchange, which involves running parallel replicas of MD simulations, we will now discuss about adaptive sampling methods.

1.5.8 Adaptive Sampling Methods

Adaptive sampling aims to improve the sampling of the configuration space by concentrating simulation efforts in regions that will improve the ensemble representation. These approaches often uses techniques like Markov State Modeling (MSM) or low dimensional reduction algorithms such as PCA to drive the simulations to unseen macrostates.
These methodologies were mainly developed for protein folding, process characterized by rare transitions events to a folded state. The observation of such rare events, which require crossing a high free energy barrier, depends on cumulative simulation time rather than the length of the simulation itself. This is because the probability of overcoming a free energy barrier is determined by the total number of attempts made at crossing it. Adaptive seeding, therefore, involves initiating simulations from regions of space likely to facilitate free energy barrier crossing. This approach improve the sampling of the conformational space by using the information acquired through previous simulations. Regions of the conformational space already explored do not provide significant new information, while new unexplored regions adds valuable insights to fully recover the full conformational space. Here we will briefly introduce a popular adaptive sampling strategy based on states decomposition by MSM.
First, we will give a brief overview of MSM. To discretize the configuration space, MSM employs states known as microstates and estimates their probability distribution as well as the probabilities of transitions between these states for a given time lag τ. MSM extract kinetic properties by employing time-lagged independent component analysis (TICA) based on kinetic proximity. Transition probabilities are collected in a transition matrix T_{ij}, which records the probabilities of transitioning from state i to state j with time lag τ. By taking into account transition probabilities, it becomes possible to construct the probability density and microstates. Subsequently, these microstates are commonly clustered into macrostates using a distance metric. In this case, the adaptive sampling algorithm aims to enhance the discovery of unseen macrostates by using one or multiple relatively short MD simulations that are launched in parallel, the initial structures of these MDs are selected from the macrostates obtained by MSM.
In this context, the aim of adaptive sampling algorithm coupled to MSM is to enhance the exploration of undiscovered macrostates revealed by MSM. This is accomplished by running one or multiple relatively short MD simulations in parallel, with the initial structures of these simulations selected from the macrostates identified by MSM. For example one of the first proposed strategy involved randomly selecting a fixed number of structures from each macrostate and was termed adaptive seeding.[135, 136] Another strategy proposed seeding simulations from states that contribute the most to the statistical uncertainty of MSMs constructed in each iteration.[137]
However, MSM rely on hyperparameters and require large simulations to obtain the macrostates. Thus, some methods do not rely on MSM analysis. iMapD [138] employs clustering in a low-dimensional manifold inferred through dimensionality reduction and selects states to seed from the

boundaries of a diffusion map in diffusion coordinates. Similarly, configurations can be chosen for reseeding using dimensionality reduction algorithms like sketch-map.[139] In Chapter 3 Section 3.1 of the thesis, an unsupervised density-driven adaptive sampling method was presented. Its primary objective was to improve the exploration of a low-dimensional space of CVs without relying on MSM or diffusion maps. This approach offers high flexibility and facilitates the development of various hybrid methods.

1.5.9 Hybrid methods

A common approach in hybrid methods is to combine enhanced sampling methods that use different principles together as some limitation of one can be circumvent by another and vice versa. One common strategy involves integrating an enhanced sampling method that biases specific degrees of freedom or CV (e.g., metadynamics or GaMD) with a method that more broadly enhances the sampling of a large number, or even all, degrees of freedom (e.g., replica exchange methods, adaptive sampling). This hybrid approach allows for improved sampling of slow orthogonal degrees of freedom that may not be adequately explored by the biased CV set alone.
For instance, it is possible to combine replica exchange and umbrella sampling, known as REUS.[140] In REUS, multiple replicas are simulated at the same temperature, but each replica has an umbrella potential centered at a distinct location. By facilitating exchanges between neighboring umbrella windows, the convergence of the sampling process is enhanced.
More recently, a method called GaREUS has been introduced,[141] which integrates REUS with GaMD. REUS enhances the sampling along predefined reaction coordinates, while GaMD accelerates conformational dynamics by incorporating a boost potential into the system potential energy. GaREUS offers more efficient sampling compared to REUS or GaMD alone, while utilizing the same computational resources as REUS.
In Chapter 3, Section 3.3, we further extended these advancements by combining our newly introduced GaMD mode, GaMD-dualwater, with umbrella sampling, along with the adaptive sampling algorithm presented in Section 3.1. The resulting ASUS-GaMD setup significantly reduces the time to convergence by factors of 10 and 20, respectively, in comparison to GaMD-US and umbrella sampling alone.
Moreover, there has been increasing efforts to overcome limitations of enhanced sampling techniques using machine learning approaches. Machine learning techniques have been instrumental in automating the definition of reaction coordinates or descriptors that accurately describe the underlying atomic systems, thereby driving advancements in the field.

1.5.10 Hybrid Machine Learning-driven Enhanced Sampling Techniques

Finding physically relevant CVs can be extremely challenging. Data-driven techniques offer a way to systematically estimate them from simulation data.[142] Most of these techniques employ unsupervised learning methods to identify low-dimensional representations of the atomic coordinates. They

can be categorized into linear and nonlinear methods. In linear, or alternatively non-linear, techniques CVs are a linear, or nonlinear, combinations of input features. Usually, nonlinear techniques can capture more complex CVs and are thus particularly suited for biomolecular simulations. However, linear techniques are often more interpretable, robust, require less data, and can incorporate nonlinearities through feature engineering.

CV discovery techniques can be categorized into two main types: those focusing on high-variance CVs and those identifying slow CVs. High-variance CVs capture the significant variance in the data when projected onto a low-dimensional space. On the other hand, slow CVs exhibit high autocorrelation and capture the long-time kinetics of the system. While slow and high-variance collective modes are often related, this is not always the case. Estimating slow CVs requires time series data, such as molecular dynamics trajectories, while high-variance CVs can be computed from temporally unordered data, such as Monte Carlo trajectories.

Next, we will briefly introduce some machine learning models used to identify these two types of CVs. Firstly, we will focus on techniques that estimate high-variance CVs, as these were the primary focus during this thesis, as they are particularly important in the context of molecular dynamics. The most famous technique for estimating high-variance CVs is PCA or kernel PCA.[143] One of the first method developed is the Molecular Enhanced Sampling with Autoencoders (MESA) [144] which is a technique that alternates between discovering nonlinear CVs using autoencoders and applying free energy biasing with umbrella sampling along these CVs. However, free energy biasing introduces a deviation from the Boltzmann distribution, which affects the loss function optimized by any model. MESA faces the challenge of optimizing a different loss at each iteration due to the changing distribution of biased simulation data. This lack of convergence poses difficulties in determining a stopping rule for the iterative process. To address this, the Free Energy Biasing and Iterative Learning with AutoEncoders (FEBILAE) incorporates a reweighting step to ensure consistency in the optimized loss and convergence of the learned CVs [145]. Another iterative method that employs a reweighting protocol is reweighted autoencoded variational Bayes for enhanced sampling (RAVE) [146]. RAVE utilizes variational autoencoders to discover nonlinear CVs. It iteratively refines the distribution of CVs by comparing it with trial CVs sampled from molecular dynamics. RAVE determines an optimal CV and probability distribution, which are used to bias a new simulation with a reweighted procedure. This process continues until thermodynamic observables reach convergence.

Recent advancements in deep reinforcement learning (DRL) have also opened up new possibilities for discovering CVs in molecular systems. DRL algorithms require the definition of a reward function, state space, and action space. In molecular dynamics simulations, the atomic coordinates can serve as the state space, atomic movements as the action space, and potential energy as the reward. One of the approach that use DRL for identifying CVs is Reinforcement learning-based adaptive sampling (REAP).[147] REAP utilizes reinforcement learning to dynamically identify the relative importance of each CV in driving the exploration of configurational space. It then adaptively initiates new simulations from configurations with high reward functions.

Now, we will present some models used to identify slow CVs. The identification of slow CVs offers valuable insights from various perspectives. Mechanistically, these CVs reveal the collective modes that dictate the metastable and transitions states of a system. They also provide information about the

system's structural, thermodynamic, and dynamic properties. In terms of enhanced sampling, they serve as suitable variables for applying biases to accelerate barrier crossing and improve configurational phase space exploration. Several approaches have been proposed for analyzing MD time series data to estimate slow CVs. These techniques approximate slow modes as linear combinations of predefined basis functions derived from the input coordinates. Examples of such techniques include time-lagged independent component analysis (tICA) [148] and Markov state models (MSM) [149, 150]. Recently, a deep learning-powered variational approach for Markov processes called VAMPnets demonstrated superior performance compared to current state-of-the-art Markov modeling techniques for studying the kinetics of molecular processes [151]. Additionally, tICA has been combined with the kernel trick to develop kernel tICA [152], which aims to approximate slow CVs using nonlinear functions of the input features. Kernel tICA has been used in conjunction with Markov state models to provide estimates of protein folding and ligand binding. Enhanced sampling can be performed using the learned slow CVs, similar to high-variance CVs. However, the use of biasing potentials perturbs the system dynamics, necessitating subsequent analysis of the biased data. Moreover, it should be noted that although slow CVs are optimal for studying rare events in some cases, such as biomolecular systems, there are situations where the identified slow CVs may have timescales that are beyond the relevant timescales of the phenomenon of interest. In such cases, corrective measures may be required to adjust the kinetic model by eliminating undesired modes.

Furthermore, machine learning can be also integrated with sampling techniques without the requirement of predicting CVs. An example of this is the deep boosted molecular dynamics (DBMD) [153], which employs probabilistic Bayesian neural network models to construct boost potentials. The boost potentials being designed to follow a Gaussian distribution with minimized anharmonicity within the context of the GaMD model.

To wrap-up, the development of machine learning-based enhanced sampling techniques is an active field of research, as it promises to drastically improve conformational space exploration in many ways, including the discovery of CVs and the distribution of potential boosts. In Chapter 3, we introduced unsupervised data-driven enhanced sampling techniques, which aim to identify high-variance CVs through PCA but can be combined with various low dimensional reduction algorithms such as autoencoders but also various enhanced sampling methods to further enhance its sampling efficiency.

2 Enhancing Force Field accuracy through Neural Networks, Long range interactions and Nuclear Quantum Effects

2.1 Advancing Accuracy in Many-Body Dispersion-Corrected Density Functional Theory: A Deep Learning-aided Density-Free Approach

Introduction

This section introduces a novel deep learning-aided dispersion model that holds promise for application in both FF and dispersion-corrected DFT. While FF relies on a simple vdW model to describe dispersion interactions, DFT inherently lacks the ability to accurately capture these interactions due to its dependence on local contributions to electronic correlation.[34] To overcome these limitations, a recent advancement has been made with the development of a range-separated linear scaling stochastic formulation of the MBD@rsSCS dispersion model. This formulation incorporates non-additive many-body dispersion (MBD) effects by self-consistently screening atomic polarizabilities, while also addressing the original computational complexity bottleneck of $\mathcal{O}(N^3)$ by employing a stochastic Lanczos trace estimator. [154, 17, 42, 155]

The model exhibits several advantageous features, including linear scaling, communication-free parallel implementations, minimal memory requirements, and the potential to compute accurate many-body interactions for complex biosystems with millions of atoms within minutes. However, it does require the partitioning of the electron density, which can be computationally intensive. Existing partitioning schemes like Hirshfeld suffer from inaccuracies in representing atomic properties.[43] To mitigate these issues, the Iterative Hirshfeld (HI) scheme has been developed, though it still faces challenges with density interpolation for negatively charged atoms. An alternative approach, the iterative Stockholder atom (ISA) scheme, overcomes these problems by employing a molecular density averaging process. The minimal basis iterative Stockholder atom (MBISA), a variant of ISA, has shown success in rescaling atomic polarizabilities and reproducing electronic structure properties.[156, 157, 158]

In this work, a hybrid deep neural network-aided MBD@rsSCS model (DNN-MBD) [15] is proposed, incorporating Atom-In-Molecule (AIM) volumes generated by a deep neural network trained on

MBISA AIM volumes from the ANI-1 dataset. This approach highlights the capability of deep learning to provide a means to bypass expensive quantum mechanical computations by capturing local atomic properties. The DNN-MBD model offers a potential solution for improving the accuracy of dispersion-corrected density functional theory, as well as general FF models. We will show that evaluation of the DNN-MBD model on the S66x8 benchmark set, coupled with common PBE/PBE0 density functionals, demonstrates its ability to model non-additive long-range dispersion interactions in systems containing millions of atoms at a low computational cost, with accuracy comparable to CCSD(T)/CBS accuracy enabling its use in generating large and extremely accurate dataset for machine learning models.[159]

THE JOURNAL OF PHYSICAL CHEMISTRY LETTERS
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

pubs.acs.org/JPCL

Letter

Accurate Deep Learning-Aided Density-Free Strategy for Many-Body Dispersion-Corrected Density Functional Theory

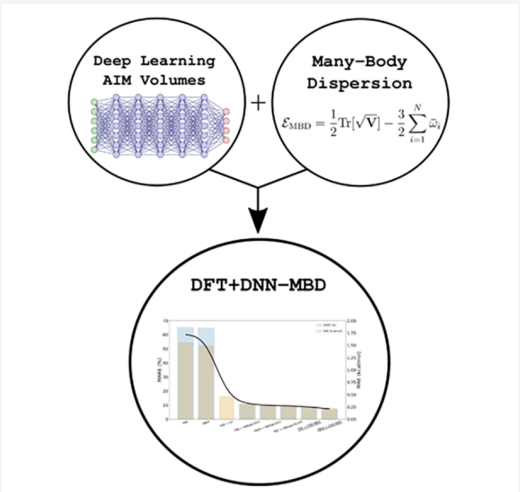
Pier Paolo Poier,* Théo Jaffrelot Inizan, Olivier Adjoua, Louis Lagardère, and Jean-Philip Piquemal*

Cite This: *J. Phys. Chem. Lett.* 2022, 13, 4381−4388

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Using a deep neuronal network (DNN) model trained on the large ANI-1 data set of small organic molecules, we propose a transferable density-free many-body dispersion (DNN-MBD) model. The DNN strategy bypasses the explicit Hirshfeld partitioning of the Kohn−Sham electron density required by MBD models to obtain the atom-in-molecules volumes used by the Tkatchenko−Scheffler polarizability rescaling. The resulting DNN-MBD model is trained with minimal basis iterative Stockholder atomic volumes and, coupled to density functional theory (DFT), exhibits comparable (if not greater) accuracy to other approaches based on different partitioning schemes. Implemented in the Tinker-HP package, the DNN-MBD model decreases the overall computational cost compared to MBD models where the explicit density partitioning is performed. Its coupling with the recently introduced Stochastic formulation of the MBD equations (*J. Chem. Theory Comput.* **2022**, *18* (3), 1633−1645) enables large routine dispersion-corrected DFT calculations at preserved accuracy. Furthermore, the DNN electron density-free features extend the MBD model's applicability beyond electronic structure theory within methodologies such as force fields and neural networks.



Since its original formulation in 1965, Kohn−Sham density functional theory[1] (KS-DFT) has become the most popular family of electronic structure methods. KS-DFT represents in fact the cheapest way for introducing electronic correlation as its computational cost is similar to that of the Hartree−Fock method. KS-DFT is based on the idea of evaluating the kinetic energy from a Slater determinant, thus assuming the electrons to be noninteracting. This apparently crude assumption actually leads to big improvements in describing chemical bonding compared to, for example, the use of the Thomas−Fermi kinetic energy formulation. The difference between the Slater determinant kinetic energy representation and the true one, together with the difference between the true total electronic interaction and the exchange energies, represents, in KS-DFT, the key contribution to the exchange-correlation functional which remains, however, unknown.

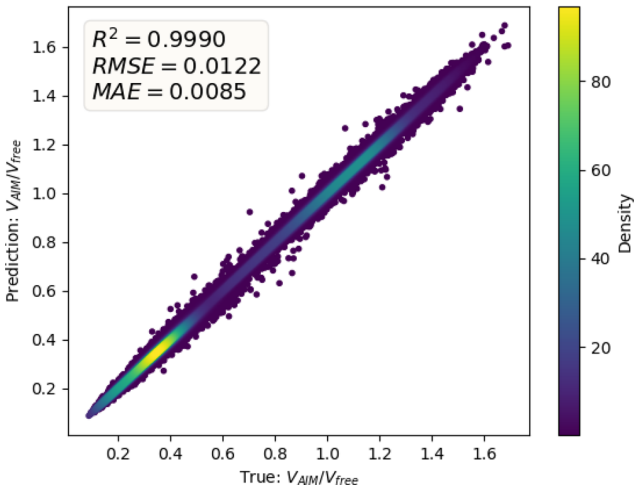


Figure 1. Atomic volume correlation plot comparing the DNN prediction to DFT reference calculations for 1/100 of the validation set. The color bar scale reflects the density of points and correlates with the atomic volume ratio distribution (Figure 2 of the SI).

In practice, the plethora of existing KS-DFT variants differentiate themselves in the way the exchange-correlation functional is approximated. Typically it is assumed to be a functional of the local electron density and eventually of its gradient and Laplacian. As a consequence, only local

Received: March 31, 2022
Accepted: May 6, 2022
Published: May 11, 2022

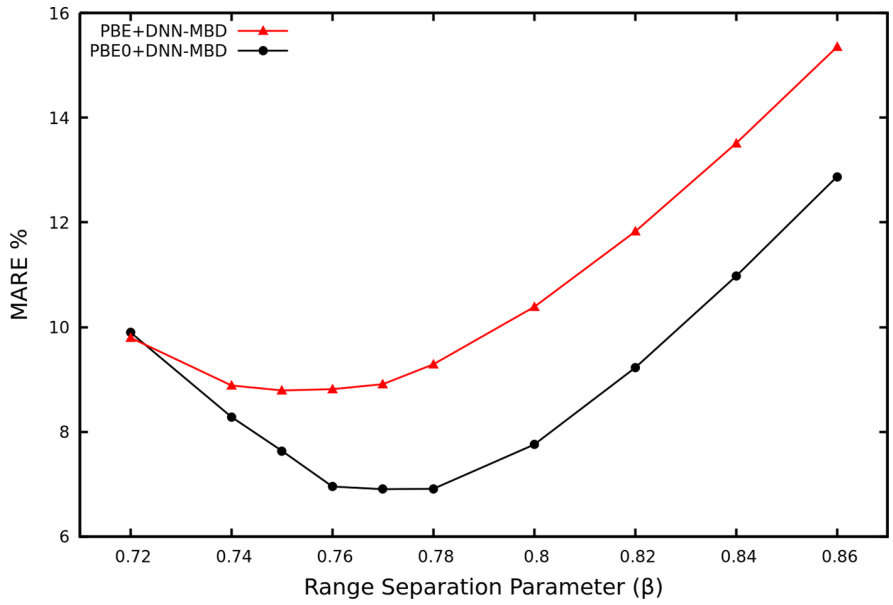


Figure 2. MARE (%) as a function of the range separation parameter for the PBE+DNN-MBD and PBE0+DNN-MBD methods.

contributions to electronic correlation are included, and this explains the general inadequacy of DFT methods to describe dispersion interactions which, on the other hand, have roots in long-range electronic correlation.

To retain the pleasant computational performances of KS-DFT methods, several dispersion corrections have been proposed.[2] Among these, the popular and successful approach of Grimme includes dispersion via empirical pairwise C_6 terms.[3−5] This is particularly appealing in virtue of its nearly zero additional computational cost.

A further approach is to replace the empirical pairwise terms with ones obtained from quantities coupled to the molecular electron density. For example, in Becke and Johnson's model, pairwise C_6 coefficients are written in terms of atomic polarizabilities and the averaged exchange-hole dipoles corresponding to each of the two atoms in the pair.[6,7] In the alternative approach proposed by Tkatchenko and Scheffler (TS),[8] pairwise C_6 coefficients are instead expressed in terms of accurate free atom reference data as well as atoms-in-molecule (AIM) polarizabilities obtained by rescaling free atom ones via AIM volumes computed via the Hirshfeld partitioning of the molecular electron density.[9]

One limitation of the above-mentioned pairwise approaches is the impossibility of capturing nonadditive many-body dispersion (MBD) effects, whose inclusion has recently been shown important in modeling extended systems, supramolecular complexes, and proteins in solutions, among others.[10−13]

The nonadditive long-range character of dispersion interactions has been modeled via a set of coupled fluctuating dipoles[14,15] (CFD) or alternatively by quantum Drude oscillators.[16−19]

In recent years, Tkatchenko, DiStasio, Ambrosetti, et al. have proposed a range-separated many-body dispersion model based on the CFD where the self-consistent screening of a set of atomic polarizabilities is performed (MBD@rsSCS).[20,21] The MBD@rsSCS model is appealing not only for introducing nonadditive many-body dispersion effects but also since it relies, *de facto*, on a single range-separation parameter which is tuned according to the choice of the exchange-correlation functional employed.

The MBD@rsSCS keeps in fact the spirit of the TS approach where AIM polarizabilities and van der Waals radii are obtained via the Hirshfeld partitioning of the density.

The Hirshfeld method leads to AIM densities which minimize the Kullback−Lieber divergence corresponding to the information loss upon molecule formation where this solid mathematical condition is used as a basis for the development of new information-theoretic partitioning methods.[22]

As discussed in refs 23 and 24, Hirshfeld partitioning makes its resulting AIM densities as close as possible to the ones of the isolated atoms; consequently, AIM's properties turn out to be as similar as possible to those of the free atoms. This is particularly evident in the magnitude of Hirshfeld atomic charges, being too small in magnitude for reproducing the molecular electrostatic potential (ESP) or in modeling AIM polarizabilities in ionic and covalent crystals where the Hirshfeld partitioning leads to unrealistically large polarizabilities of cations which can even be found to be larger than those of the anions.[25]

The above-mentioned shortcomings were ameliorated by the iterative Hirshfeld (HI) scheme[26] where the reference atomic density employed in the partitioning is constructed as a linear combination of the two densities relative to the atomic oxidation states closest to the fractional number of electrons assigned by the partitioning at a given iteration.

The ESPs computed from HI atomic charges have proven to agree remarkably well with the *ab initio* computed reference.[27] In addition, the use of HI derived AIM polarizabilities leads to more realistic dispersion coefficients,[25] especially in ionic systems and adsorption phenomena on surfaces of ionic solids where the HI scheme used within the TS dispersion model improves interaction energies.[28] HI partitioning has also been employed in the MBD@rsSCS model, replacing the original Hirshfeld scheme,[29] and its use in connection to the fractionally ionic AIM polarizabilities leads, in the just mentioned challenging systems, to reduced errors.[30]

Despite the improvements gained by the HI partitioning, the scheme remains affected by a shortcoming arising from the density interpolation for negatively charged atoms, as this procedure is, for some species, ill-defined. This arises from the

https://doi.org/10.1021/acs.jpclett.2c00936
J. Phys. Chem. Lett. 2022, 13, 4381−4388

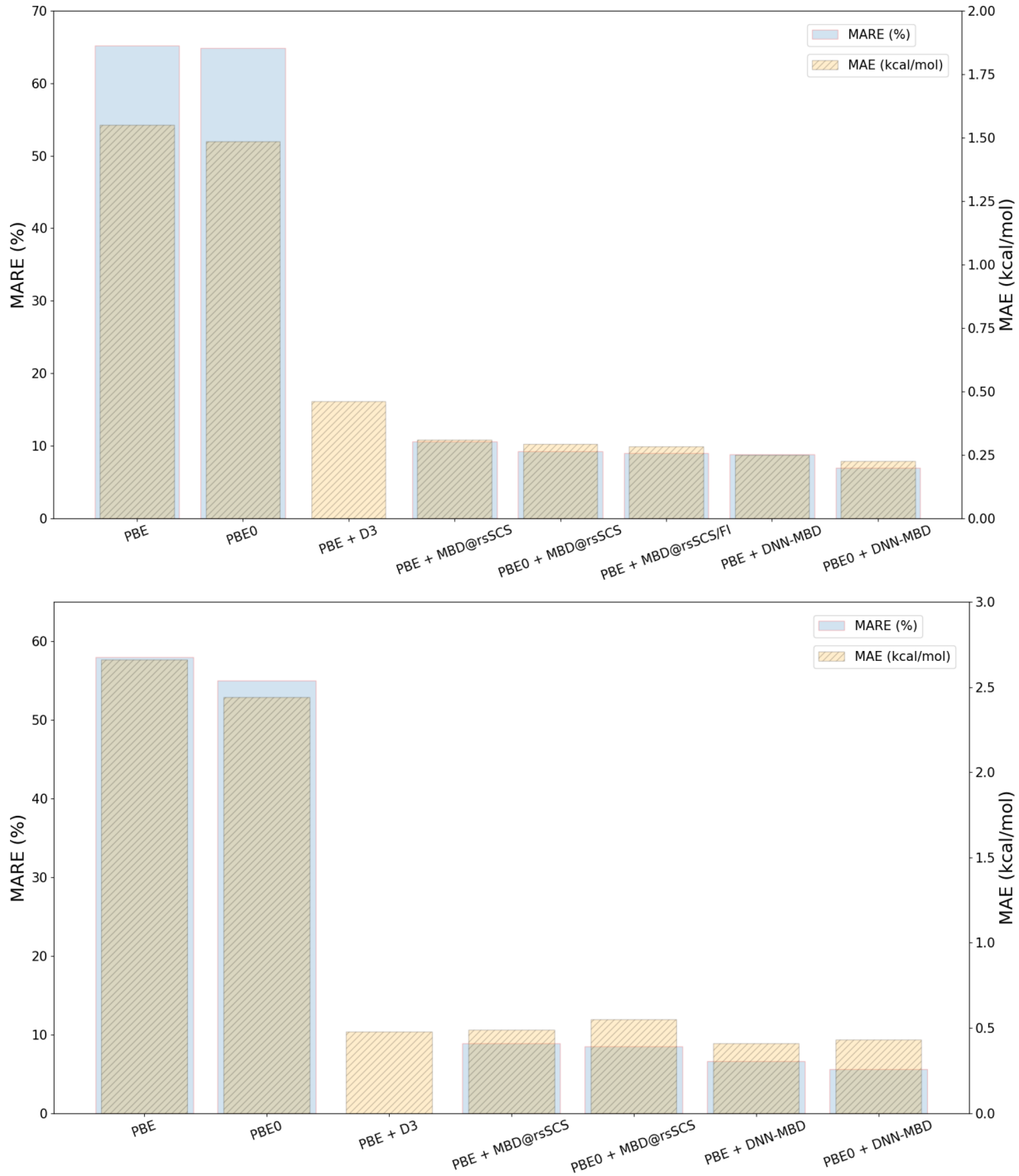


Figure 3. MARE (%) and MAE (kcal/mol) of PBE, PBE0, PBE+D3,[56] and different MBD models (MBD@rsSCS[21] and MBD@rsSCS/FI[30]) including our DNN-MBD for the S66x8 (top) and S22 (bottom) data sets.

fact that free anions such as N^- and O^{2-} (or in general any doubly negative ion) are not bound and their reference electron densities, computed at a complete basis set (CBS), result in a detached electron.

The iterative Stockholder atom (ISA) scheme, on the other hand, is not affected by this problem as the partitioning does not require reference atomic densities computed from isolated atoms at different ionic states, as they are rather obtained from a spherical averaging of the molecular density using nuclei as expansion points.[31,32] The minimal basis iterative Stockholder atom (MBISA), a variant of the ISA method, has proven successful in the atomic polarizability rescaling approach employed by the TS scheme as well as in reproducing *ab initio* ESP from atomic point charges,[33] and for this reason its use in connection to the MBD@rsSCS model is particularly appealing.

AIM properties are local quantities which depend on the near chemical environment and thus carry a certain degree of transferability. In particular, the TS polarizability rescaling scheme (employed in the MBD@rsSCS) makes use of AIM volumes, which are well suited to be computed via deep neural

https://doi.org/10.1021/acs.jpclett.2c00936
J. Phys. Chem. Lett. 2022, 13, 4381−4388

Table 1. MAE (kcal/mol) and MARE (%) Relative to the S66x8 Data Set for Our DNN-Based Models as well as for a Few Other Dispersion Correction Ones[a]

model	MAE [kcal/mol]	MARE [%]
PBE	1.55	65
PBE0	1.48	65
PBE+D3	0.44	n.a.
PBE+MBD@rsSCS (β = 0.83)	0.32	10.6
PBE0+MBD@rsSCS (β = 0.85)	0.30	9.2
PBE+MBD@rsSCS/FI (β = 0.83)	0.28	9.0
PBE+DNN-MBD (β = 0.75)	0.25	9.0
PBE0+DNN-MBD (β = 0.77)	0.23	6.9

[a]For the MBD-based models, the method-specific range separation parameter reported in parentheses refers to the one optimized for the S66x6 set. MAE and MARE are computed taking revised CCSD(T) CBS energies.

Table 2. MAE (kcal/mol) and MARE (%) Relative to the S22 Data Set for Our DNN-Based Models as well as for a Few Other Dispersion Correction Ones[a]

model	MAE [kcal/mol]	MARE [%]
PBE	2.66	58
PBE0	2.44	55
PBE0+MBD@rsSCS (β = 0.85)	0.55	8.5
PBE+MBD@rsSCS (β = 0.83)	0.49	8.9
PBE+D3	0.48	n.a.
PBE0+DNN-MBD (β = 0.77)	0.43	5.6
PBE+DNN-MBD (β = 0.75)	0.41	6.6

[a]For the MBD-based models, the method-specific range separation parameter reported in parentheses refers to the one optimized for the S66x6 set. MAE and MARE are computed taking revised S22 energies where, compared to the original paper, a larger basis set was employed.[58].

network (DNN) where the environment vector associated with an atom's surrounding is defined within a local cutoff. The potential of deep learning in capturing local atomic properties has been proved by Isayev and co-workers, whose multioutput DNN model successfully predicts AIM properties ranging from multipoles to volumes.[34]

In this letter, we present a hybrid DNN-aided MBD@rsSCS (DNN-MBD) model where the AIM volume ratio employed in the TS polarizability rescaling is generated by a deep neural network trained on the ANI-1 data set (approximately 4.6 million structures) containing MBISA AIM volumes.[35]

For the common S66x8 benchmark set,[36] the DNN-MBD model coupled to the common PBE/PBE0 density functionals exhibits excellent interaction energies while completely bypassing the electron density partitioning with a consequent computational cost reduction. This electron density-free DNN-MBD approach employed in connection to our recently proposed linear scaling stochastic MBD@rsSCS formulation[37] allows for modeling nonadditive long-range dispersion interactions of up-to-millions of atoms systems at a very low computational cost without compromising the accuracy.

We note that kernel-ridge regression approaches to model AIM polarizabilities have been proposed in modeling dispersion interactions.[38,39] This approach, however, is characterized by a $O(N^2)$ and $O(N^3)$ scaling of the required memory and computational cost involved in the model's training, with N being the size of the data set. This nonlinear scaling prevents the applicability of kernel-ridge approaches on very large and diverse data sets, necessary for the generation of general-purpose MBD models. Additionally, the poor scaling with the number of processes limits its use on large systems. Here instead we generalize the approach to model MBD interactions to a much broader class of systems thanks to the employed model's flexibility and broad data set, without affecting the model's accuracy and linear scalability.

We will, in the following, proceed by briefly recalling the key concepts of the standard MBD@rsSCS model before introducing the DNN-MBD hybrid model and its performances.

As a starting point in this discussion, we examine the TS polarizability rescaling in eq 1, where α_i and V_i represent the TS static polarizability and AIM volume, respectively, of the ith atom, while the zero superscript denotes free atom reference quantities.

$$\alpha_i = \left(\frac{V_i}{V_i^0}\right)\alpha_i^0 \tag{1}$$

The AIM volume V_i is obtained by solving the integral in eq 2, where $\rho(\mathbf{r})$ is the Kohn−Sham molecular electron density, which, via the partitioning-specific weight function $w_i(\mathbf{r})$, is decomposed into its AIM densities $\{\rho_i(\mathbf{r})\}$.

$$V_i = \int \mathbf{r}^3 \rho_i(\mathbf{r})\, \mathrm{d}\mathbf{r}$$
$$\rho_i(\mathbf{r}) = w_i(\mathbf{r})\rho(\mathbf{r}) \tag{2}$$

Once the set of static AIM polarizabilities in eq 1 is obtained, a corresponding set of frequency-dependent ones is generated via eq 3, where this time ω_j^0 and $C_{6,j}^0$ are the free atom characteristic excitation frequency and first dispersion coefficient.

$$\alpha_j(i\nu) = \frac{\alpha_j}{1 - (i\nu/\omega_j^0)^2}$$
$$\omega_j^0 = \frac{4}{3}\frac{C_{6,j}^0}{(\alpha_j^0)^2} \tag{3}$$

These frequency-dependent polarizabilities are, in the MBD@rsSCS model, gathered as diagonal elements of the frequency-dependent superpolarizability matrix $\mathbf{A}(i\nu)$, which is one of the entries in the Dyson-like equation below whose solution provides the screened superpolarizability matrix $\bar{\mathbf{A}}(i\nu)$.

$$\bar{\mathbf{A}}(i\nu) = \mathbf{A}(i\nu) - \mathbf{A}(i\nu)\mathbf{T}^{\mathrm{SR}}(i\nu)\bar{\mathbf{A}}(i\nu) \tag{4}$$

$\mathbf{T}^{\mathrm{SR}}$ represents a damped dipole−dipole interaction operator applied to the Coulombic interaction of two frequency-dependent spherical Gaussian charge distributions, where its explicit expression, together with the one for $\mathbf{A}(i\nu)$, can be found in ref 37. We note here that the Fermi damping function employed in the definition of $\mathbf{T}^{\mathrm{SR}}$ makes use of AIM van der Waals radii, which can also be obtained by a volume rescaling similarly to what was discussed for polarizabilities.[8]

The solution of eq 4 for a set of frequencies, and a consequent partial contraction of the converged $\{\bar{\mathbf{A}}(i\nu)\}$, gives a set of screened frequency-dependent atomic polarizabilities $\{\bar{\alpha}_j(i\nu)\}$ which are used to approximate the Casimir−Polder integral providing screened characteristic excitation frequencies $\{\bar{\omega}_j\}$.

https://doi.org/10.1021/acs.jpclett.2c00936
J. Phys. Chem. Lett. 2022, 13, 4381−4388

$$\bar{C}_{6,j} = \frac{3}{\pi}\int_0^\infty \bar{\alpha}_j(i\nu)\bar{\alpha}_j(i\nu)\,\mathrm{d}\nu$$

$$\bar{\omega}_j = \frac{4}{3}\frac{\bar{C}_{6,j}}{[\bar{\alpha}_j(0)]^2} \tag{5}$$

The set of screened excitation frequencies as well as the screened static atomic polarizabilities define the MBD potential matrix shown in eq 6 for a general ij block. $\mathbf{T}^{\mathrm{LR}}$ represents the range-separated damped dipole−dipole interaction matrix, whose explicit expression is also found in ref 37.

$$\mathbf{V}_{ij} = \delta_{ij}\bar{\omega}_i^2 + (1-\delta_{ij})\bar{\omega}_i\bar{\omega}_j\sqrt{\bar{\alpha}_i(0)\bar{\alpha}_j(0)}\,\mathbf{T}_{ij}^{\mathrm{LR}} \tag{6}$$

The trace of $\sqrt{\mathbf{V}}$ defines the interaction energy $\mathcal{E}_{\mathrm{int}}$ of the CFDs in the system,[37] while its zero-point value $\mathcal{E}_0$ is given by the sum of all screened excitation frequencies. Finally, the difference between $\mathcal{E}_{\mathrm{int}}$ and $\mathcal{E}_0$ gives the target MBD@rsSCS energy, eq 7, which is coupled to the KS-DFT one to include nonadditive dispersion contributions.

$$\mathcal{E}_{\mathrm{MBD}} = \mathcal{E}_{\mathrm{int}} - \mathcal{E}_0 = \frac{1}{2}\mathrm{Tr}[\sqrt{\mathbf{V}}] - \frac{3}{2}\sum_{i=1}^{N}\bar{\omega}_i \tag{7}$$

In the original MBD@rsSCS model just briefly reviewed, $\mathcal{E}_{\mathrm{MBD}}$ is coupled to the molecular electron density via AIM volume partitioning introduced in eq 2.

In this letter, instead we show that the explicit electron density partitioning can be avoided by learning AIM volumes via a DNN model without affecting the original MBD@rsSCS model's accuracy.

Bereau et al. and, more recently, Mulhi et al. used Machine Learning on atomic volumes inside the vdW model to capture the many body effects.[38,39] Both have developed a Gaussian approximation potential (GAP) force field on TS polarizability rescaling. While GAP has been shown to outperform neural networks in predicting energies with a small-sized data set, e.g., a few thousand data points, its poor computational scaling $O(N^3)$ prevents its use on very large training sets and thus to build a general purpose MBD model.[40] Finally, these models are either restricted to pairwise interactions or do not scale linearly with respect to the number of atoms as our stochastic reformulation of the MBD equations was introduced only recently.[37]

Isayev et al.[34] recently extended their 5 million chemical conformations, the ANI-1 data set, with atomic volumes computed at the ωB97x/def2-TZVPP level with MBISA partitioning. In virtue of its size and diversity, this data set is here employed in building our DNN to be coupled to the MBD@rsSCS model. Here we restrict ourselves to structures composed of only C, H, N, and O, thus reducing the actual data set size to 4.6 million conformations.

In the MBISA weight function $w_i(\mathbf{r})$, each of the reference pro-atomic densities $\rho_i^0(\mathbf{r})$ is expanded into m_i Slater functions, m_i being the number of shells of atom i placed at $\mathbf{R}_i$.

$$w_i(\mathbf{r}) = \frac{\rho_i^0(\mathbf{r})}{\sum_{j=1}^{N}\rho_j^0(\mathbf{r})}$$

$$\rho_i^0(\mathbf{r}) = \sum_{\sigma=1}^{m_i}\frac{N_{i,\sigma}}{k_{i,\sigma}^3 8\pi}\exp\left(-\frac{\|\mathbf{r}-\mathbf{R}_i\|}{k_{i,\sigma}}\right) \tag{8}$$

In the scheme, the population $N_{i,\sigma}$ and width $k_{i,\sigma}$ of each shell are free variables which are optimized so that the loss of information upon molecule formation is minimized.[33] To handle such a large data set, a deep neural network is the natural choice.[41] In particular, we use as a machine learning model a feed-forward DNN with the ANI-like symmetry functions (SFs).[42] The ANI's SFs are a subfamily of Behler−Parinello's ones,[43] which traduce an atomic local environment i into an atomic environment vector (AEV) $G_i = \{G_i^R, G_i^A\}$ where G_i^R and G_i^A represent its radial and angular contributions, respectively. Although SF development is an intensive field of research and more accurate models have been developed since (ωACSF,[44] SOAP,[45] among others), we stick to the ANI's original SFs as they were shown to successfully predict complex local properties such as, in the case of AIMNET, multipoles and volumes.[34] Moreover, ANI's SFs have the great advantage of being computationally efficient as they rely on 2-body terms, thus making the overall DNN model linear scaling with the system's size.

The DNN part of the combined DNN-MBD model relies on Scikit-learn,[46] PyTorch,[47] and TorchAni.[48] They are all included in the Tinker-HP neural network module, whose implementation will be detailed in a forthcoming dedicated paper (T. Jaffrelot Inizan et al., 2022).

We kept the original ANI's SF parameters as we did not see major differences after tuning them. We empirically tested multiple neural network architectures (further details are found in the Supporting Information (SI) Figure 3), and the best performance was obtained with five hidden layers. The atomic element's neural network architectures are H 160:128:96:48:1; C 144:112:96:48:1; N 128:112:96:48:1; O 128:112:96:48:1. We observed that adding one extra layer to the original ANI-1x model architecture slightly increases the performance of the model while making it more flexible. Indeed, in the original ANI-1x model, the last layer is composed of 96 neurons, and adding an extra 48 neurons layer may prevent loss of information. We used the Exponential Linear Units (ELU) activation function[49] while the model's parameters were initialized with the so-called "He" initialization and updated with Hutter's AdamW algorithm during the training procedure.[50] Within the AdamW algorithm, the factor was set to 0.5 and the patience to 100. The initial learning rate was set to 10^{-3}, and the early stopping learning rate was set to 10^{-6}. The ANI-1 data set was shuffled and split into a training and validation set containing 80% and 20%, respectively, of the full data set. The networks were trained for 6000 epochs with a batch size of 2560.

The ANI-1 data set, upon which our DNN model is trained, consists of AIM volumes computed at the ωB97x/def2-TZVPP level. The model is trained on volume ratios rather than pure AIM volumes as the narrower distribution of the former allows for a DNN's better performance without the need for rescaling. Indeed, the atomic volumes ratio for C, H, O, and N (see Figure 1 of the SI) is between 0.1 and 1.6. Free atom volumes are computed at the same level as AIM ones. The correlation plots between the DNN model and the *ab initio* validation set reference is depicted in Figure 1. The root-mean-square error (RMSE) and mean-absolute error (MAE) are, respectively, 0.012 and 0.008, which is much less than the smallest value of the data set showing the good accuracy of our model. The final DNN model and the data set used for the training can be downloaded directly via the Zenodo repository located at the URL in ref 51.

The DNN model providing the AIM volume ratios is embedded in the Tinker-HP package where our linear-scaling

https://doi.org/10.1021/acs.jpclett.2c00936
J. Phys. Chem. Lett. 2022, 13, 4381−4388

and embarrassingly parallel stochastic MBD@rsSCS is also implemented.[37]

The resulting DNN-MBD model is coupled to the common semilocal PBE[52] functional as well as its hybrid PBE0 version[53] since this choice allows for comparisons with results ready available in the literature. The optimal range-separation β parameters for both the PBE+DNN-MBD and PBE0+DNN-MBD methods are obtained by minimizing the mean absolute relative error (MARE) on the widely employed S66x8 benchmark set consisting of 66 dimers placed at 8 different intermolecular distances for a total of 528 different structures where CCSD(T) interaction energies computed at CBS are used as reference.

All DFT computations employed Jensen's pcseg-3 basis set belonging to the family of segmented polarization-consistent[54] basis sets which, for DFT calculations, exhibit lower basis set errors than other Gaussian basis sets as well as higher computational efficiency at a given cardinal number, as these basis sets were explicitly designed and optimized for DFT.[55]

Figure 2 shows the MARE as a function of the range separation parameter for PBE+DNN-MBD and PBE0+DNN-MBD methods.

The optimal β parameters are found to be 0.75 and 0.77 for the PBE+DNN-MBD and PBE0+DNN-MBD methods, respectively. These values differ from the ones optimized for the original PBE/PBE0+MBD@rsSCS models,[21] and this has to be addressed with the different partitioning scheme employed. As pointed out by Vestraelen et al.,[33] AIM densities computed via the Hirshfeld or HI partitioning exhibit asymmetries; i.e., they are aspherical with too much density in the bonding region. This density accumulation, relatively far away from the atomic nucleus, leads to larger values of radial moments, thus leading to larger AIM volumes compared to the ones obtained via the MBISA scheme (unaffected from this asymmetry artifact) for which less screening of volume-scaled AIM quantities (smaller β) is most likely to be needed.[33]

We observe, nevertheless, that both PBE0+DNN-MBD and PBE0+MBD@rsSCS methods require a larger β parameter compared to their PBE corresponding models, and this is consistent with the PBE0 improved description of short-range exchange-correlation effects due to the fraction of exact exchange included in the functional, as discussed in ref 21.

The performances of the optimized PBE/PBE0+DNN-MBD methods is compared to those of different MBD models in terms of MAE and MARE for the S66x8 data set, and the results are summarized in Figure 3 with actual values reported in Table 1.

For the benchmark set here employed, the DNN-MBD model exhibits lower (although by a contained margin) errors both in its coupling to the PBE and PBE0 functionals compared to the standard MBD@rsSCS approach based on Hirshfeld AIM volumes as well the PBE+MBD@rsSCS/FI approach based on the fractionally ionic polarizabilities and HI AIM volume partitioning. For both the chosen functionals, the outcoming DNN-MBD model provides a mean absolute error in the S66x8 interaction energies, which is below 0.25 kcal/mol compared to the reference CCSD(T) CBS golden standard.

To strengthen the analysis, we additionally computed the MAE and MARE for the S22 data set[57] by employing the range separation parameters previously optimized for the S66x8 set. We can, in this way, employ the S22 set as a test set to validate our conclusions, Figure 3 (bottom) and Table 2.

Compared to the S66x8 set, the MAE and MARE values of our proposed PBE/PBE0+DNN-MBD models are, for the S22 set, higher; however, this is not surprising as no β optimization was performed this time. Let us note that all methods present errors that are larger in the case of the S22 set compared to S66x8 (see Table 1 and Table 2). Indeed, there are reasons for that, and we can stress that the dimers employed in the S22 set are placed at equilibrium while the S66x8 set includes out of equilibrium dimers. In our case, the DNN-MBD model trained on S66x8 appears less biased toward equilibrium structures. Overall, as one can see from Table 2, our DNN-MBD model remains highly transferable and, with an error below 0.43 kcal/mol compared to the reference CCSD(T) CBS gold standard, outperforms the previous S22 results obtained with others methods.

Having been trained on a large and diverse set of AIM volumes, the resulting DNN-MBD model inherits the strengths of the MBISA scheme discussed earlier in this letter while completely bypassing the explicit density partitioning with a consequent decrease of the computational cost. We also note that the DNN model could be successfully trained with different AIM partitioning schemes due to the locality of the target quantities (volumes).

The presented density-free DNN-SMBD model is included in the Tinker-HP package[59] and will be released with the next version of the software. There, it can benefit from the linear-scaling embarrassingly parallel performances of our stochastic formulation (SMBD) of the MBD key equations, whose remarkable computational performances have been recently discussed.[37]

We believe that the present DNN-SMBD model can be beneficial in applications of dispersion-corrected DFT to large complex systems requiring an accurate yet extremely efficient inclusion of MBD effects. The DNN model, by avoiding the direct solution of the KS equations due to its electron density-free features, allows for the ready application of the DNN-SMBD approach in the development of accurate *ab initio*-based force fields[60,61] and neural network methodologies.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpclett.2c00936.

Training plots for DNN with different layers (PDF)

Raw PBE, PBE+DNN-MBD, PBE0, and PBE0+DNN-MBD energies for the S66x8 data set (ZIP)

Raw PBE, PBE+DNN-MBD, PBE0, and PBE0+DNN-MBD energies for the S22 data set (ZIP)

■ AUTHOR INFORMATION

Corresponding Authors

Pier Paolo Poier − *Sorbonne Université, LCT, UMR 7616 CNRS, Paris 75005, France;* orcid.org/0000-0003-0907-7242; Email: pier.poier@sorbonne-universite.fr

Jean-Philip Piquemal − *Sorbonne Université, LCT, UMR 7616 CNRS, Paris 75005, France; Department of Biomedical Engineering, The University of Texas at Austin, Austin, Texas 78713, United States;* orcid.org/0000-0001-6615-9426; Email: jean-philip.piquemal@sorbonne-universite.fr

Authors

Théo Jaffrelot Inizan − *Sorbonne Université, LCT, UMR 7616 CNRS, Paris 75005, France*

Olivier Adjoua − *Sorbonne Université, LCT, UMR 7616 CNRS, Paris 75005, France*

Louis Lagardère − *Sorbonne Université, LCT, UMR 7616 CNRS, Paris 75005, France; Sorbonne Université, IP2CT, FR 2622 CNRS, Paris 75005, France*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jpclett.2c00936

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation program (Grant Agreement No. 810367), project EMC2 (J.-P.P.). Computations have been performed at GENCI (IDRIS, Orsay, France, and TGCC, Bruyres le Chatel) under Grant No. A0070707671.

■ REFERENCES

(1) Kohn, W.; Sham, L. J. Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **1965**, *140*, A1133−A1138.
(2) Grimme, S.; Hansen, A.; Brandenburg, J. G.; Bannwarth, C. Dispersion-corrected mean-field electronic structure methods. *Chem. Rev.* **2016**, *116*, 5105−5154.
(3) Grimme, S. Accurate description of van der Waals complexes by density functional theory including empirical corrections. *J. Comput. Chem.* **2004**, *25*, 1463−1473.
(4) Grimme, S. Semiempirical GGA-type density functional constructed with a long-range dispersion correction. *J. Comput. Chem.* **2006**, *27*, 1787−1799.
(5) Grimme, S.; Antony, J.; Ehrlich, S.; Krieg, H. A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu. *J. Chem. Phys.* **2010**, *132*, 154104.
(6) Johnson, E. R.; Becke, A. D. A post-Hartree-Fock model of intermolecular interactions: inclusion of higher-order corrections. *J. Chem. Phys.* **2006**, *124*, 174104.
(7) Becke, A. D.; Johnson, E. R. Exchange-hole dipole moment and the dispersion interaction revisited. *J. Chem. Phys.* **2007**, *127*, 154108.
(8) Tkatchenko, A.; Scheffler, M. Accurate molecular van der Waals interactions from ground-state electron density and free-atom reference data. *Phys. Rev. Lett.* **2009**, *102*, 073005.
(9) Hirshfeld, F. L. Bonded-atom fragments for describing molecular charge densities. *Theoretica chimica acta* **1977**, *44*, 129−138.
(10) Reilly, A. M.; Tkatchenko, A. Seamless and accurate modeling of organic molecular materials. *J. Phys. Chem. Lett.* **2013**, *4*, 1028−1033.
(11) Ambrosetti, A.; Alfè, D.; DiStasio, R. A.; Tkatchenko, A. Hard numbers for large molecules: toward exact energetics for supramolecular systems. *J. Phys. Chem. Lett.* **2014**, *5*, 849−855.
(12) Ambrosetti, A.; Ferri, N.; DiStasio, R. A.; Tkatchenko, A. Wavelike charge density fluctuations and van der Waals interactions at the nanoscale. *Science* **2016**, *351*, 1171−1176.
(13) Stöhr, M.; Tkatchenko, A. Quantum mechanics of proteins in explicit water: the role of plasmon-like solute-solvent interactions. *Science Advances* **2019**, *5*, eaax0024.
(14) Langbein, D. Microscopic calculation of macroscopic dispersion energy. *J. Phys. Chem. Solids* **1971**, *32*, 133−138.
(15) Donchev, A. G. Many-body effects of dispersion interaction. *J. Chem. Phys.* **2006**, *125*, 074713.
(16) Sommerfeld, T.; Jordan, K. D. Quantum Drude Oscillator Model for describing the interaction of rxcess electrons with water clusters: an application to (H2O)13-. *J. Phys. Chem. A* **2005**, *109*, 11531−11538.
(17) Jones, A. Quantum drude oscillators for accurate many-body intermolecular forces. Ph.D. thesis, University of Edinburgh, 2010.
(18) Jones, A. P.; Crain, J.; Sokhan, V. P.; Whitfield, T. W.; Martyna, G. J. Quantum Drude oscillator model of atoms and molecules: many-body polarization and dispersion interactions for atomistic simulation. *Phys. Rev. B* **2013**, *87*, 144103.
(19) Odbadrakh, T. T.; Jordan, K. D. Dispersion dipoles for coupled Drude oscillators. *J. Chem. Phys.* **2016**, *144*, 034111.
(20) Tkatchenko, A.; DiStasio, R. A.; Car, R.; Scheffler, M. Accurate and efficient method for many-body van der Waals interactions. *Phys. Rev. Lett.* **2012**, *108*, 236402.
(21) Ambrosetti, A.; Reilly, A. M.; DiStasio, R. A.; Tkatchenko, A. Long-range correlation energy calculated from coupled atomic response functions. *J. Chem. Phys.* **2014**, *140*, 18A508.
(22) Heidar-Zadeh, F.; Ayers, P. W.; Verstraelen, T.; Vinogradov, I.; Vöhringer-Martinez, E.; Bultinck, P. Information-theoretic approaches to atoms-in-molecules: Hirshfeld family of partitioning schemes. *J. Phys. Chem. A* **2018**, *122*, 4219−4245.
(23) Ayers, P. W. Atoms in molecules, an axiomatic approach. I. Maximum transferability. *J. Chem. Phys.* **2000**, *113*, 10886−10898.
(24) Ayers, P. W.; Morrison, R. C.; Roy, R. K. Variational principles for describing chemical reactions: condensed reactivity indices. *J. Chem. Phys.* **2002**, *116*, 8731−8744.
(25) Bučko, T.; Lebègue, S.; Ángyán, J. G.; Hafner, J. Extending the applicability of the Tkatchenko-Scheffler dispersion correction via iterative Hirshfeld partitioning. *J. Chem. Phys.* **2014**, *141*, 034114.
(26) Bultinck, P.; Van Alsenoy, C.; Ayers, P. W.; Carbó-Dorca, R. Critical analysis and extension of the Hirshfeld atoms in molecules. *J. Chem. Phys.* **2007**, *126*, 144111.
(27) Van Damme, S.; Bultinck, P.; Fias, S. Electrostatic potentials from self-consistent Hirshfeld atomic charges. *J. Chem. Theory Comput.* **2009**, *5*, 334−340.
(28) Bučko, T.; Lebègue, S.; Hafner, J.; Ángyán, J. G. Improved density dependent correction for the description of London dispersion forces. *J. Chem. Theory Comput.* **2013**, *9*, 4293−4299.
(29) Deringer, V. L.; Csányi, G. Many-body dispersion correction effects on bulk and surface properties of rutile and anatase TiO2. *J. Phys. Chem. C* **2016**, *120*, 21552−21560.
(30) Gould, T.; Lebègue, S.; Ángyán, J. G.; Bučko, T. A fractionally ionic approach to polarizability and van der Waals many-body dispersion calculations. *J. Chem. Theory Comput.* **2016**, *12*, 5920−5930.
(31) Lillestolen, T. C.; Wheatley, R. J. Atomic charge densities generated using an iterative stockholder procedure. *J. Chem. Phys.* **2009**, *131*, 144101.
(32) Misquitta, A. J.; Stone, A. J.; Fazeli, F. Distributed Multipoles from a Robust Basis-Space Implementation of the Iterated Stockholder Atoms Procedure. *J. Chem. Theory Comput.* **2014**, *10*, 5405−5418.
(33) Verstraelen, T.; Vandenbrande, S.; Heidar-Zadeh, F.; Vanduyfhuys, L.; Van Speybroeck, V.; Waroquier, M.; Ayers, P. W. Minimal basis iterative stockholder: atoms in molecules for force-field development. *J. Chem. Theory Comput.* **2016**, *12*, 3894−3912.
(34) Zubatyuk, R.; Smith, J. S.; Leszczynski, J.; Isayev, O. Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network. *Science Advances* **2019**, *5*, eaav6490.
(35) Smith, J. S.; Zubatyuk, R.; Nebgen, B.; Lubbers, N.; Barros, K.; Roitberg, A. E.; Isayev, O.; Tretiak, S. The ANI-1ccx and ANI-1x data sets, coupled-cluster and density functional theory properties for molecules. *Scientific Data* **2020**, *7*, 134.
(36) Řezáč, J.; Riley, K. E.; Hobza, P. S66: A Well-balanced database of benchmark interaction energies relevant to biomolecular structures. *J. Chem. Theory Comput.* **2011**, *7*, 2427−2438.
(37) Poier, P. P.; Lagardère, L.; Piquemal, J.-P. O(N) stochastic evaluation of many-body van der Waals energies in large complex systems. *J. Chem. Theory Comput.* **2022**, *18*, 1633−1645.
(38) Bereau, T.; DiStasio, R. A.; Tkatchenko, A.; von Lilienfeld, O. A. Non-covalent interactions across organic and biological subsets of chemical space: physics-based potentials parametrized from machine learning. *J. Chem. Phys.* **2018**, *148*, 241706.
(39) Muhli, H.; Chen, X.; Bartók, A. P.; Hernández-León, P.; Csányi, G.; Ala-Nissila, T.; Caro, M. A. Machine learning force fields based on local parametrization of dispersion interactions: Application to the phase diagram of C_{60}. *Phys. Rev. B* **2021**, *104*, 054106.
(40) Zuo, Y.; Chen, C.; Li, X.; Deng, Z.; Chen, Y.; Behler, J.; Csányi, G.; Shapeev, A. V.; Thompson, A. P.; Wood, M. A.; Ong, S. P.

Performance and cost assessment of machine learning interatomic potentials. *J. Phys. Chem. A* **2020**, *124*, 731−745.
(41) Westermayr, J.; Chaudhuri, S.; Jeindl, A.; Hofmann, O. T.; Maurer, R. J. Long-range dispersion-inclusive machine learning potentials for structure search and optimization of hybrid organic-inorganic interfaces. *arXiv*, 2022; https://arxiv.org/abs/2202.13009.
(42) Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.* **2017**, *8*, 3192−3203.
(43) Behler, J.; Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.
(44) Gastegger, M.; Schwiedrzik, L.; Bittermann, M.; Berzsenyi, F.; Marquetand, P. wACSF−Weighted atom-centered symmetry functions as descriptors in machine learning potentials. *J. Chem. Phys.* **2018**, *148*, 241709.
(45) Bartók, A. P.; Kondor, R.; Csányi, G. On representing chemical environments. *Phys. Rev. B* **2013**, *87*, 184115.
(46) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: machine learning in Python. *Journal of machine learning research* **2011**, *12*, 2825−2830.
(47) Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. Presented at NIPS 2017 Workshop on Autodiff, 2017.
(48) Gao, X.; Ramezanghorbani, F.; Isayev, O.; Smith, J. S.; Roitberg, A. E. TorchANI: A free and open source PyTorch-based deep learning implementation of the ANI neural network potentials. *J. Chem. Inf. Model.* **2020**, *60*, 3408−3415.
(49) Clevert, D.-A.; Unterthiner, T.; Hochreiter, S. Fast and accurate deep network learning by exponential linear units (ELUs). *arXiv*, 2015; https://arxiv.org/abs/1511.07289.
(50) Loshchilov, I.; Hutter, F.; Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv*, 2017; https://arxiv.org/abs/1711.05101.
(51) Poier, P. P.; Jaffrelot Inizan, T.; Adjoua, O.; Lagardère, L.; Piquemal, J.-P. ANI-1 dataset with added atomic volume ratios restricted to CHNO atoms for DNN-MBD; DOI: 10.5281/zenodo.6397639.
(52) Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **1996**, *77*, 3865−3868.
(53) Adamo, C.; Barone, V. Toward reliable density functional methods without adjustable parameters: The PBE0 model. *J. Chem. Phys.* **1999**, *110*, 6158−6170.
(54) Jensen, F. Polarization consistent basis sets: principles. *J. Chem. Phys.* **2001**, *115*, 9113−9125.
(55) Jensen, F. Unifying general and segmented contracted basis sets. *Segmented polarization consistent basis sets. Journal of Chemical Theory and Computation* **2014**, *10*, 1074−1085.
(56) Goerigk, L.; Kruse, H.; Grimme, S. Benchmarking Density Functional Methods against the S66 and S66x8 Datasets for Non-Covalent Interactions. *ChemPhysChem* **2011**, *12*, 3421−3433.
(57) Jurečka, P.; Šponer, J.; Černý, J.; Hobza, P. Benchmark database of accurate (MP2 and CCSD(T) complete basis set limit) interaction energies of small model complexes, DNA base pairs, and amino acid pairs. *Phys. Chem. Chem. Phys.* **2006**, *8*, 1985−1993.
(58) Takatani, T.; Hohenstein, E. G.; Malagoli, M.; Marshall, M. S.; Sherrill, C. D. Basis set consistent revision of the S22 test set of noncovalent interaction energies. *J. Chem. Phys.* **2010**, *132*, 144104.
(59) Lagardère, L.; Jolly, L.; Lipparini, F.; Aviat, F.; Stamm, B.; Jing, Z. F.; Harger, M.; Torabifard, H.; Cisneros, G. A.; Schnieders, M. J.; Gresh, N.; Maday, Y.; Ren, P. Y.; Ponder, J. W.; Piquemal, J. P. Tinker-HP: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields. *Chem. Sci.* **2018**, *9*, 956−972.
(60) Gresh, N.; Cisneros, G. A.; Darden, T. A.; Piquemal, J.-P. Anisotropic, Polarizable Molecular Mechanics Studies of Inter- and Intramolecular Interactions and Ligand-Macromolecule Complexes. A Bottom-Up Strategy. *J. Chem. Theory Comput.* **2007**, *3*, 1960−1986.
(61) Naseem-Khan, S.; Lagardère, L.; Narth, C.; Cisneros, G. A.; Ren, P.; Gresh, N.; Piquemal, J.-P. Development of the Quantum Inspired SIBFA Many-Body Polarizable Force Field: Enabling Condensed Phase Molecular Dynamics Simulations. *J. Chem. Theory. Comput.* **2022**, DOI: 10.1021/acs.jctc.2c00029.

Conclusion

The developed DNN-MBD model, trained with MBISA volumes and coupled with DFT, demonstrates an improved accuracy compared to other approaches relying on different partitioning schemes. By implementing it in the Tinker-HP package, the DNN-MBD model significantly reduces the overall computational cost compared to MBD models that involve explicit density partitioning. The combination of the DNN-MBD model with the recently introduced Stochastic formulation of the MBD equations enables efficient and accurate dispersion-corrected DFT calculations on a large scale. Moreover, as pointed out previously, the electron density-free characteristics of the DNN-MBD model extend its applicability beyond electronic structure theory to methodologies such as FFs and neural networks. This versatility makes the DNN-MBD model highly valuable for simulations on large and complex systems, where the inclusion of MBD effects is crucial. By circumventing the need to directly solve the equations for computing the electrostatic potential, the DNN model enables the straightforward application of the DNN-MBD approach in the development of accurate *ab-initio*-based FFs and neural network methodologies, as demonstrated by its accuracy, which is comparable to CCSD(T)/CBS with an error of only 0.25 kcal/mol.[160, 161]

These advancements pave the way for generating extensive and highly accurate datasets for future machine learning models, offering new avenues for research and development in the field. The DNN-MBD model holds great promise in enhancing the computational efficiency and accuracy of dispersion-corrected DFT. Its flexibility enable also its integration with various methods ranging from FF to electronic structure methods.

2.2 Integrating Hybrid Deep Neural Networks, Polarizable Force Fields and quantum-accurate Long-Range Effects with the multi-GPU Deep-HP platform

Introduction

Recently, there have been significant advancements in MLPs, enhancing their transferability and scalability across a wide range of systems. However, their applications have largely been limited to studying small chemical systems, remaining far away from biological modeling. Additionally, the lack of efficient multi-GPU infrastructure software for MLPs within existing molecular dynamics packages has been a challenge. Molecular dynamics software such as Tinker-HP is predominantly coded in Fortran, C++, and CUDA, while ML libraries primarily use Python through libraries like PyTorch and TensorFlow, making their efficient integration cumbersome. This work aims to bridge this gap by introducing a highly efficient GPU-resident platform, Deep-HP, within the Tinker-HP package.

Furthermore, this research seeks to address some limitations of MLPs. Indeed, the inherent architecture of MLPs constrains them to short-range interactions. Although recent developments have shown that MLPs can capture long-range charge transfer and multiple charge states, their computational cost remains considerably higher compared to physics-based PFFs models. Additionally, they struggle to accurately describe solute behavior in water environments as well as water model.[19, 105]

To overcome these challenges, this section presents Deep-HP, a multi-GPU MLP platform integrated into Tinker-HP, enabling the coupling and development of MLPs with state-of-the-art many-body polarizable effects. Tinker-HP utilizes 3D decomposition for massive parallelization, which aligns well with the approach often employed in MLP development, decomposing the total energy into atomic energy contributions. The scalability of the platform with MLPs is theoretically linear, allowing for scaling to hundreds or thousands of GPUs for large systems.

In this section, the scalability and implementation of Deep-HP on the ANI model, one of the most accurate MLPs for small organic molecules to date, are extensively tested.[102] Additionally, in line with quantum mechanics/molecular mechanics embedding simulations, a hybrid MLP/molecular mechanics strategy is introduced. This strategy leverages the ANI-MLP potential for solute-solute interactions while utilizing the AMOEBA PFF to evaluate solvent-solute and solvent-solvent interactions. This combination allows ANI-MLP to benefit from AMOEBA's strengths, including accurate flexible water and protein models in condensed phases, the incorporation of counter-ions, and the inclusion of long-range and many-body effects. The performance of the model is evaluated through calculations of solvation free energies for 80 molecules in four organic solvents, as well as binding free energies for 14 challenging host-guest complexes from SAMPL blind challenges.[162]

Chemical Science

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE

View Article Online
View Journal

Cite this: DOI: 10.1039/d2sc04815a

All publication charges for this article have been paid for by the Royal Society of Chemistry

Scalable hybrid deep neural networks/polarizable potentials biomolecular simulations including long-range effects†

Théo Jaffrelot Inizan,[a] Thomas Plé,[a] Olivier Adjoua,[a] Pengyu Ren,[b] Hatice Gökcan,[c] Olexandr Isayev,[c] Louis Lagardère[ad] and Jean-Philip Piquemal*[ab]

Received 30th August 2022
Accepted 3rd April 2023

DOI: 10.1039/d2sc04815a

rsc.li/chemical-science

Deep-HP is a scalable extension of the Tinker-HP multi-GPU molecular dynamics (MD) package enabling the use of Pytorch/TensorFlow Deep Neural Network (DNN) models. Deep-HP increases DNNs' MD capabilities by orders of magnitude offering access to ns simulations for 100k-atom biosystems while offering the possibility of coupling DNNs to any classical (FFs) and many-body polarizable (PFFs) force fields. It allows therefore the introduction of the ANI-2X/AMOEBA hybrid polarizable potential designed for ligand binding studies where solvent–solvent and solvent–solute interactions are computed with the AMOEBA PFF while solute–solute ones are computed by the ANI-2X DNN. ANI-2X/AMOEBA explicitly includes AMOEBA's physical long-range interactions *via* an efficient Particle Mesh Ewald implementation while preserving ANI-2X's solute short-range quantum mechanical accuracy. The DNN/PFF partition can be user-defined allowing for hybrid simulations to include key ingredients of biosimulation such as polarizable solvents, polarizable counter ions, *etc.*... ANI-2X/AMOEBA is accelerated using a multiple-timestep strategy focusing on the model's contributions to low-frequency modes of nuclear forces. It primarily evaluates AMOEBA forces while including ANI-2X ones only *via* correction-steps resulting in an order of magnitude acceleration over standard Velocity Verlet integration. Simulating more than 10 μs, we compute charged/uncharged ligand solvation free energies in 4 solvents, and absolute binding free energies of host–guest complexes from SAMPL challenges. ANI-2X/AMOEBA average errors are discussed in terms of statistical uncertainty and appear in the range of chemical accuracy compared to experiment. The availability of the Deep-HP computational platform opens the path towards large-scale hybrid DNN simulations, at force-field cost, in biophysics and drug discovery.

1 Introduction

Understanding the dynamics of biological systems is of prime importance in structural biology and drug discovery. Over the last 50 years, coupled to force fields (FFs), molecular dynamics (MD) simulations have proven to be an essential theoretical tool to predict the long-timescale behaviour of proteins in complex environments. In recent years, deep learning technologies have also progressed and showed some potential to accelerate drug discovery. For example, DeepMind developed the Alphafold[2] (ref. 1) model that is able to predict over 200 million protein structures. Proteins' properties could, however, drastically change during a molecular dynamics simulation. For instance, the protein–water interface can drive fluctuations of catalytic cavities and thus change drug inhibition. MD is therefore the prominent approach to go beyond simple structures in order to predict the complete protein conformational space.[2–4] Due to the biological system sizes and biological simulation timescales, pure quantum chemistry models cannot be used for simulations and are replaced by empirical FFs, which are presently commonly used to model chemical interactions.

FFs model the total energy as a sum over intra and intermolecular energy terms. The treatment of the latter leads to two classes of FFs: classical and polarizable. In classical FFs, the intermolecular interactions are modeled by Lennard-Jones potential and Coulomb potential which make them computationally efficient enabling modern software to tackle long timescale simulation of complex systems.[5–8] While offering

[a]Sorbonne Université, Laboratoire de Chimie Théorique, UMR 7616 CNRS, Paris, 75005, France. E-mail: jean-philip.piquemal@sorbonne-universite.fr
[b]Department of Biomedical Engineering, University of Texas at Austin, Austin, Texas, USA
[c]Department of Chemistry, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA
[d]Sorbonne Université, Institut Parisien de Chimie Physique et Théorique, FR 2622 CNRS, Paris, France

† Electronic supplementary information (ESI) available. See DOI: https://doi.org/10.1039/d2sc04815a

reasonable precision thanks to careful parametrization,[9,10] classical FFs lack an accurate description of polarization and to a larger extent of many-body physical effects.[11,12] These quantities can play a crucial role in solvation[2,3] and in the stability of secondary and quaternary structures of proteins.[12] The development of polarizable FFs (PFFs) has opened new routes able to explicitly include many-body effects.[13,14] Their computational cost has long hindered their use but with the rise of High Performance Computing (HPC)[15,16] and the increasing performance of computational devices such as GPUs, million-atom PFF simulations are now possible.[17]

At this stage, Machine Learning (ML) schemes also have the potential to offer a new paradigm for boosting MD simulations and to play their role in the development of FFs. ML potentials (MLPs) also avoid solving the Schrödinger equation at each time-step of the simulation by providing a mathematical direct relationship between the atomic positions and the potential energy. In recent years, MLPs have been an active field of research which led to the emergence of different frameworks such as high-dimensional deep neural network potentials (HDNNPs), Gaussian approximation potentials,[18] moment tensor potentials, spectral neighbor analysis potentials,[19] atomic cluster expansion, graph networks, kernel ridge regression methods,[20] gradient-domain machine learning[21–24] and support vector machines.[25] MLP nonlinear functional forms are very general and highly flexible, allowing for a very accurate representation of electronic structure computation reference data. The input of an MLP is usually hand-crafted real valued functions of the coordinates that preserve some symmetries and uniquely defined atomic environments. In practice, the choice of this descriptor is central to designing an accurate MLP. A variety of physics-based descriptors have been developed such as the smooth overlap of atomic positions,[26] the spectrum of approximated Hamiltonian matrix representations,[27] the Coulomb matrix and the atom-centered symmetry functions.[28,29] The latter, introduced by Behler and Parinello in 2007, is still the most popular descriptor used for HDNNP and has been employed in numerous studies.[28,30] It describes the atomic environment of a given central atom inside a cutoff radius R_c by the use of radial and angular functions. Some modifications of the initial symmetry functions have been done since, aiming to reduce the number of symmetry functions that exhibit quadratic growth with the number of elements or improve the probing of the atomic environment.[31] However, even if such descriptors have considerably improved the transferability and the scalability of HDNNPs, they are often used to only study small chemical systems that remain far away from the needs of biological modeling. They have nevertheless already been shown to be useful to create buffer region neural network in QM/MM (Quantum Mechanics/Molecular Mechanics) simulations to minimize overpolarization artifacts of the QM region due to classical MM.[32] Another issue has been the lack of efficient MLP multi-GPU infrastructure software inside an already existing molecular dynamics package. In the last couple of years things started to change and our work is part of this large movement and also aims to address the recent development of the ML-field.[33] While our work aims to utilize new developments in the ML-field, we also aim to address some of the shortcomings of MLPs. Indeed, the intrinsic architecture of MLP usually constrains them to short-range interactions. Recently, Tsz Wai Ko *et al.* proposed a fourth-generation of HDNNP which is able to capture long-range charge transfer and multiple charge states.[34] While it demonstrates the power of ML, its computational cost is much higher compared to physics-based PFF long-range models and is not yet able to correctly describe a solute in water.

To address these challenges, we present Deep-HP (HP stands for High-Performance), a multi-GPU MLP platform which is part of the Tinker-HP package and enables the coupling and development of MLPs with state-of-the-art many-body polarizable effects. Tinker-HP uses massive parallelization by means of 3D decomposition which is a particularly well suited strategy for MLPs that are often developed by decomposing the total energy as a sum of atomic energy contributions.[15,17] The platform theoretical scalability with MLPs is linear and allows scaling up to hundreds/thousands of GPUs on large systems. As the present code shares the Tinker-HP capabilities, it allows for invoking fast physics-based many-body energy contributions. We extensively test Deep-HP scalability and implementation on the ANI model, one of the most accurate MLPs to date for small organic molecules. Finally, in the spirit of polarizable QM/MM embedding simulations,[35–37] we introduce a hybrid DNN/MM strategy that uses the ANI DNN to model solute–solute interactions and the AMOEBA PFF to evaluate solvent–solute and solvent–solvent interactions. This enables ANI to benefit from AMOEBA's strengths that include an accurate condensed phase flexible water and protein model, and the capability to include counter-ions and long-range/many-body effects. It should increase ANI transferability to a broader range of systems including charged ones. The performance of the model is evaluated by calculating the solvation free energies of various molecules in four organic solvents as well as the binding free energies of 14 challenging host–guest complexes taken from SAMPL blind challenges.

2 Method

2.1 Potential energy models

2.1.1 The AMOEBA polarizable force field. The total potential energy of the AMOEBA[38,39] polarizable model is expressed as the sum of bonded and non-bonded energy terms:

$$\begin{aligned} E_{\text{total}} &= E_{\text{bonded}} + E_{\text{non-bonded}} \\ E_{\text{bonded}} &= E_{\text{bond}} + E_{\text{angle}} + E_{\text{b}\theta} \\ E_{\text{non-bonded}} &= E_{\text{vdW}} + E_{\text{ele}}^{\text{perm}} + E_{\text{ele}}^{\text{pol}} \end{aligned} \quad (1)$$

The bonded terms embody MM3-like[40] anharmonic bond-stretching and angle-bending terms. Regarding the specific case of the polarizable AMOEBA water model, the intramolecular geometry and vibrations are described with a Urey–Bradley approach.[38]

The non-bonded terms include van der Waals interactions and electrostatic contributions from both permanent and

induced dipoles (polarization). More precisely, the polarization contribution is computed using an Applequist/Thole model[41] whereas Halgren's buffered 14–7 pair potential is used to model van der Waals interactions.[42] Computing the polarization energy requires the resolution of a linear system to get the induced dipoles, which is made through the use of iterative solvers such as a preconditioned conjugated gradient that is the one used in this paper (with a 10^{-5} tolerance).[17]

To model the electrostatic interactions, AMOEBA relies on point atomic multipoles truncated at the quadrupole level. More details about the functional form and parametrization of AMOEBA can be found in ref. [43] Electrostatics and many-body polarization long-range interactions are fully included through the use of the Smooth Particle Mesh Ewald approach[44,45] that allows for efficient simulations in periodic boundary conditions with $n(\log(n))$ scaling. Besides water,[38] AMOEBA is a general force field available for the biomolecular simulations of many solvents,[46] ions,[47,48] proteins[49] and nucleic acids.[50]

2.1.2 Neural network potentials. Feed-forward neural network (FFNN) is a machine learning model that uses as building blocks connected layers of nodes (*i.e.*, neurons) each associated with their weights and bias. The output of each neuron is computed through a function of the output of the previous layer. Each weight is the strength associated with a specific node connection and they are updated during the training process. The depth (*i.e.*, number of layers) of the FFNN is related to its flexibility and the complexity of the training dataset. Through careful optimization of hyperparameters, weights, biases and architecture, the FFNN can learn high dimensional non-linear functions such as potential energy surfaces. For HDNNP, the FFNN maps molecular structures to potential energy. The original HDNNP, introduced by Behler and Parrinello, expresses the total energy of a system E_T as a sum of atomic contributions E_i.

$$E_T = \sum_{i}^{N_{atoms}} E_i(G_i) \tag{2}$$

where G_i is the atomic environment vector (AEV) of atom i. Based on the assumption of locality, each atom i is associated with an AEV which probes specific radial and angular chemical regions. Each G_i is then used as the input into a single HDNNP. The construction of AEVs for each atom in the system enables the use of models for large systems even though they are trained on small molecules. Moreover, this summation has the advantage that it scales linearly with respect to the number of atoms. This atomic decomposition scheme has notably accelerated the development of HDNNP with increasingly complex architecture and AEV schemes.

2.1.3 ANI models. Smith *et al.* developed ANI, a model that uses a modified version of the Behler–Parinello symmetry functions.[31,51] Symmetry functions are building blocks of the so-called AEV, $G_i = \{G_1^X, \ldots, G_M^X\}$, which aims to probe the angular and radial local environment of a central atom i with atomic number X. The locality approximation is achieved by using a differentiable cutoff function:

$$f_c(R_{ij}) = \begin{cases} 0 & R_{ij} > R_c \\ \frac{1}{2}\cos\left\{\frac{\pi R_{ij}}{R_c}\right\} + 0.5 & R_{ij} \le R_c \end{cases} \tag{3}$$

where R_{ij} is the distance between the central atom i and a neighbor j, and R_c a cutoff radius, here fixed to 5.2 Å. To probe the neighboring environment of the central atom inside the cutoff sphere, the AEV is divided into two types of symmetry functions: radial and angular.

The commonly used radial function is a sum of products of Gaussian and cutoff functions as introduced by Behler–Parinello:

$$G_{i,m}^{rad} = \sum_{j \neq i}^{N_{atoms} \in R_c} e^{-\eta\left(R_{ij} - R_s\right)^2} f_c\left(R_{ij}\right) \tag{4}$$

The index m is associated with a set of parameters $\{\eta, R_s\}$, where R_s is the distance from the central atom for which the center of the Gaussian is shifted and η is the spatial extension of the Gaussian.

The radial symmetry functions are not sufficient to distinguish between chemical environments, *e.g.*, if the neighboring atoms are all at the same distance from atom i. This is solved by using angular symmetry functions,

$$G_{\text{ANI-ang}}^{i,m} = 2^{1-\xi} \sum_{j,k \neq i}^{N_{atoms}} \left(1 + \cos\left(\theta_{ijk} - \theta_s\right)\right)^{\xi} e^{-\eta\left(\frac{R_{ij} + R_{ik}}{2} - R_s\right)^2} f_c\left(R_{ij}\right) f_c\left(R_{ik}\right) \tag{5}$$

where θ_{ijk} is the angle between the central atom i and neighbors j and k, θ_s is used to center the maxima of the cosine and ξ changes the width of the peak. To differentiate between atom species, ANI supplied a radial part for each atomic number and an angular part for each corresponding pair inside the cutoff sphere R_c. Thus, for N atom species, the AEV has N radial and $\frac{N(N+1)}{2}$ angular sub-AEVs.

The first ANI potential, ANI-1X,[52,53] has been developed for simulating organic molecules containing H, C, N, and O chemical elements. The recent extension to ANI, ANI-2X,[54] has been trained to three additional chemical elements (S, F, and Cl). This model extends the capabilities of ANI towards more diverse chemical structures such as proteins that often contain sulfur and chlorine atoms.[54]

As ANI is mainly designed to study the dynamics of small-to medium-size organic molecules, it had not been initially coupled to a massively parallel infrastructure. In contrast, another popular MLP, introduced by Car and collaborators,[55,56] DeePMD has been pushed towards large scale simulations of millions of atoms but has been trained on some specific systems, limiting its transferability.

2.1.4 DeePMD models. The specificity of DeePMD compared to other MLPs is that it does not use hand-crafted symmetry functions to get the atomic environment.[55,56]

For an atom i, its j neighbors within a cutoff radius are first sorted according to their chemical species and their inverse distances to the central atom.

The central atom is then associated with its local frame (e_x, e_y, e_z) and the local coordinates of its neighbors are denoted as x_{ij}, y_{ij}, z_{ij}. The local environment of atom i $\{D_{ij}\}$ is then defined as:

$$\{D_{ij}\} = \left\{\frac{1}{R_{ij}}, \frac{x_{ij}}{R_{ij}}, \frac{y_{ij}}{R_{ij}}, \frac{z_{ij}}{R_{ij}}\right\} \quad (6)$$

$\{D_{ij}\}$ is then used as input for an FFNN to predict the atomic energy E_i.

DeePMD has been recently pushed in order to simulate tens of millions atoms for water and copper using a highly optimized GPU code on the Summit supercomputer[33] but it would hugely benefit from all the available features of Tinker-HP in order to run large scale biological simulations.

2.1.5 Hybrid model: neural network solutes in AMOEBA polarizable solvent/protein. Hybrid DNN/MM simulations using classical FFs have been introduced by Lahey and Rowley.[57] One technical issue with hybrid DNN/FF approaches is that in local MLP models such as ANI and DeePMD, each atom only interacts with its closest neighbors within a relatively small cutoff radius. Therefore, a correct description of long-range interactions is crucial for the simulation of condensed-phase systems, making them particularly challenging for MLP models.[58] On the other hand, particular attention has been paid during the AMOEBA parametrization to accurately reproduce condensed-phase properties of solvents (and in particular of liquid water). It is then very attractive to combine both models in order to benefit from the best of both worlds getting the small molecule quantum mechanical quality of ANI while maintaining the robustness of AMOEBA for condensed phase simulations. This can be achieved by writing the total potential energy of the so-called ANI-2X/AMOEBA hybrid model as

$$\begin{aligned} V_{\mathrm{HYB}}(P \cup W) &= V_{\mathrm{AMOEBA}}(P \cup W) + V_{\mathrm{ML}}(P) - V_{\mathrm{AMOEBA}}(P) \\ &= V_{\mathrm{AMOEBA}}(W) + V_{\mathrm{AMOEBA}}(P \cap W) \\ &\quad + V_{\mathrm{ML}}(P) \end{aligned} \quad (7)$$

where P indicates the solute, W indicates the solvent, $P \cap W$ indicates the solute–solvent interactions and $P \cup W$ indicates the total system. The many-body nature of the polarization energy prevents us from directly computing $V_{\mathrm{AMOEBA}}(P \cap W)$. To embed the ML potential, we subtract the AMOEBA potential of the isolated solute to the full AMOEBA potential. As indicated in eqn (7), this is essentially equivalent to using AMOEBA for the solvent–solvent and solvent–solute interactions and the ML model for the solute–solute interactions. The atomic environments that are given to the ML potential therefore only comprise atoms from the solute and should be similar to data present in the training set, thus reducing occurrences of extrapolation. This coupling with AMOEBA allows simulation of atom types not available with MLPs and inclusion of counter ions that are crucial in biology. This also enables the use of the accurate AMOEBA water model while benefiting from the automatic inclusion of long-range effects *via* AMOEBA's efficient Particle Mesh Ewald periodic boundary conditions.

2.2 Deep-HP: a multi-GPU MLP platform within Tinker-HP

2.2.1 A general machine learning platform. New ML architecture is introduced daily and dedicated machine learning libraries, PyTorch, TensorFlow and Keras, have created a large community of developers and users.[59–61]

Conversely, most of the MD codes (CHARMM, GROMACS, Tinker-HP,…)[7,62] are often written using compiled languages such as Fortran or C/C++. To allow for the simultaneous execution of both Python-based MLP codes and Tinker-HP we implemented an interface that allows for efficient data exchanges between environments while maintaining Tinker-HP as the master process which, punctually, calls the MLP code. Identified by Tinker-HP as another computational subroutine, the MLP code should be therefore provided as a Python API. We have implemented such functionality using the C Foreign Function Interface (cffi) for Python which allows for efficient API embedding, within a dynamic library to be linked with. Technically, within such a framework we can now call Python frozen codes from C using such cffi embedding features, thus enabling the use of various MLP codes within Tinker-HP.

In that context, the recent GPU-accelerated version of Tinker-HP[17] offers the opportunity to build an overall very efficient hybrid MD/MLP code as both applications are running on the same GPU platform. To do so, we need to design a Python/C interface in a way that avoids any substantial data transfers between Python and C environments. In practice, the cffi module is not natively designed to interface data structures from device memory: its dictionary can only process host addresses on array datatype or scalar data structures. Based on these constraints, our code would be forced to perform two host-device data transfers in order to communicate through Fortran/C and Python interface. To overcome this issue that would be detrimental to the global performance, we directly send generic memory addresses through the interface as scalar values and use the PyCUDA python module to manually cast these addresses into Tensor type that can actually be used by MLP codes. Fortunately, PyCUDA and PyTorch provide such casting routines. Thus, calling Python codes from Fortran/C with device data among the calling arguments can be done independent of the size of those arguments.

Furthermore, we built the interface of the MLP code in order to keep Tinker-HP model-agnostic. In practice, Tinker-HP provides positions and neighbor lists and gets energies and forces in return. Adding a new MLP to the platform then becomes an easy task, especially if it was developed using the PyTorch or TensorFlow libraries. Moreover, we implemented an API within TorchANI which allows us to save and reconstruct ANI-like models using JSON, YAML and PKL formats. This allows us to directly use models trained with TorchANI with the Deep-HP platform, thus reducing the hassle of transferring a model from the training stage to production simulations.

2.2.2 Massive parallelism within Tinker-HP: scalable neural network simulations. Regarding parallelism, Tinker-HP uses a three-dimensional domain decomposition (DD) scheme. The simulation box is decomposed into a certain number of domains matching the exact number of parallel processes at our disposal so that each process – attached or not to a device – is assigned to a unique domain. Then, each process computes partial forces on the local atoms, communicates the partial data to its spatial neighbors, sums the partial forces and integrates the equations of motions for local atoms at each time-step. The DD method is valid and effective under the assumption that all interactions are short-range and the atomic positions do not move much between two time-steps. The same structure has been used during the development of the accelerated multi-GPU version.[17] Naturally, we wanted to preserve this property with the MLP code interface despite the fact that TorchANI is not designed to run on multiple GPUs. Using the DD method from Tinker-HP, we can isolate the local atoms of a domain and its neighbors and send the information to an MLP code instance through the interface for calculation. We also bypass the implemented neighbor list within TorchANI, and use the one of Tinker-HP. Indeed, we verified that the TorchANI neighbor list algorithm scales as $\mathcal{O}(N^2)$ (N being the number of atoms), both in execution time and memory, which limits its applicability to small systems. For instance, a 12 000 atom water box on a Quadro GV100 GPU card supported by 32 GB memory already caused a memory overflow. Because TorchANI requires a pair list of indices as a data structure, we adapted the highly GPU-optimized linked-cell method, thoroughly described in ref. 17. In practice, the list is built by partitioning the box into smaller ones and resorting to an adjacency matrix and a filtering process. Finally, the complexity of the neighbor list generation outperforms the original TorchANI implementation, thus significantly reducing both the computational cost and memory footprint and allowing the handling of much larger systems. For example, systems made of more than 100 000 atoms are now manageable on a single 32 GB GV100 GPU. On top of that, we also noticed a constant memory allocation from Python (especially when running in parallel) which happens to be detrimental to the performance and, on some occasions, can lead to a crash. This issue has been solved by resorting to an upstream bounded buffer reservation whose size is proportional to the number of atoms in the system. In the end, Deep-HP is able to perform simulations of several million atom systems, as illustrated in Fig. 1 where we show the scalability of the platform on water boxes up to 7.7 million atoms using up to 68 V100 GPUs.

2.3 Performance and scalability results

2.3.1 Benchmark systems. We use water boxes of increasing size as benchmark systems as well as some solvated proteins.[15,17] The solvated proteins and their respective number of atoms, in parentheses, are: DHFR protein (23 558), SARS-CoV2 M^{pro} protein (98 500) and COX protein (174 219). For the water boxes: 648 (*i.e.*, small), 4800 (big), 12 000 (huge), 19 200 (globe), 96 000 (puddle), 288 000 (pond), 864 000 (lake), 2 592 000 (bay) and 7 776 000 (sea). After equilibration, we evaluated the performance on short *NVE* MD simulations.

2.3.2 GPU performances. To ensure the performance and portability of our platform, we ran tests on different GPU infrastructures such as Tesla V100 nodes of the Jean-Zay supercomputer, the Irène Joliot Curie ATOS Sequana supercomputer V100 partition or a NVIDIA DGX A100 node. In the rest of the text the default device is the Tesla V100 if not mentioned otherwise. For each system, we performed 2.5 ps MD simulations with a Verlet integrator using a 0.5 fs time-step and averaged the performance over the complete runs. Fig. 1 gathers single GPU device performances.

Before discussing performance results let us introduce three critical concepts: saturation, utilization and peak performance. Saturation represents the ratio of resources used by the algorithm against the actual resources supplied by the GPU. It is closely related to the degree of parallelism expressed within the algorithm and its practical use in the simulation. Given the fact that recent GPUs provide and execute several thousands of threads at the same time to run calculations on numerous

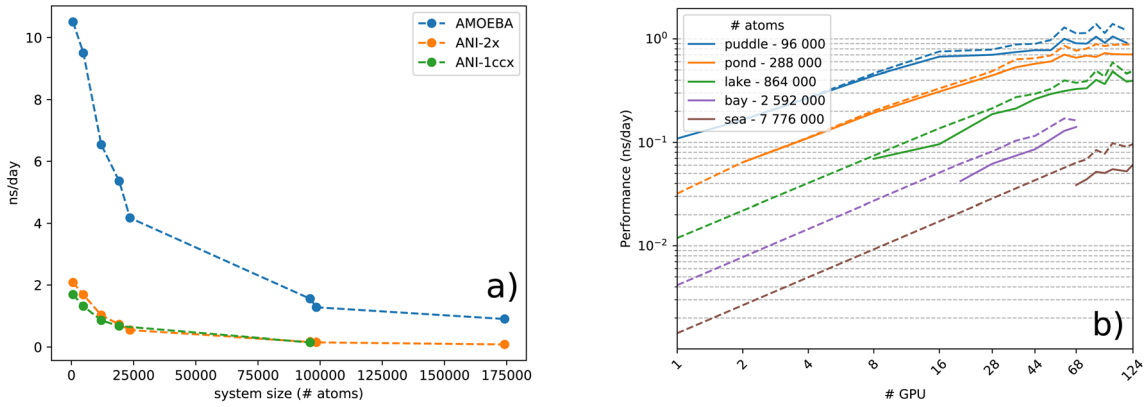


Fig. 1 (a) Performance comparison between ANI-1ccx(1NN), ANI-2X(1NN) and AMOEBA models in ns per day, over increasing system size, on a single Nvidia Tesla A100. (b) Strong scaling logarithmic scale plot of the ANI-2X model on benchmark systems. Simulations are performed in the *NVE* ensemble using a Velocity-Verlet integrator 0.2 fs time-step.

computational cores, complete saturation is naturally not achieved for small systems. On the other hand, the device utilization represents the percentage of execution time during which the GPU is active. As the GPU is driven by the CPU, its utilization heavily depends on both the CPU speed and the amount of code actually offloaded to the device. It is essential to rely on asynchronous computation and to develop a device-resident application in order to achieve a complete GPU utilization over time. Finally, peak performance (PP) describes how an algorithm asymptotically harnesses the computational power of the device on which it operates. Increasing this metric implies maximization of arithmetic operations over memory. However, one can only assess device peak performance in terms of floating point operations when both saturation and utilization are maximized. With a typical HPC device such as Quadro GV100 which delivers over 15.6 TFlop per s in single precision arithmetic (4 bytes), around 69 arithmetic operations can be performed between two consecutive float transactions from global memory, in order to reach the peak performance. Knowing this, we analyze the GPU peak performance of Deep-HP and Tinker-HP AMOEBA, in both separate and hybrid runs, using the reference GV100 card. Results are depicted in Table 1. We can see the influence of device saturation on peak performance while running pure ML models, from the under-saturated DHFR system to the over-saturated COX one. MLPs manage to achieve excellent peak performance on GPU platforms due to the large amount of calculations induced by the numerous matrix-vector products involved. For AMOEBA, on the other hand, the relatively tiny increase of peak performance for both systems – second column of Table 1 – denotes an excellent saturation and utilization of the device, regardless of the size. The overall peak, however, reaches a lower 10.52%, which is still satisfactory given the complexity of the algorithm involved in the PFF calculation.

To study the complexity of the algorithm, we ran the benchmark systems on a single DGX A100 with two ANI models and compared the performance against the AMOEBA force field (see Fig. 1a). The ANI-1ccx simulations are performed on water boxes ranging from 648 to 96 000 atoms. For ANI-2X we also considered three solvated proteins: DHFR, SARS-CoV2 M^{pro} and COX. Furthermore, for these tests, we performed inference using only one instance from the ensemble of eight neural network predictors of the ANI models. On water boxes, ANI-1ccx is found to be between 2% and 7% faster than ANI-2X due to the model's intrinsic complexities. Fig. 1a shows the performance of both ANI-2X and AMOEBA. In the 648 and 4800 atom systems, AMOEBA is 1.85 and 2.20 times faster than ANI respectively. In the first four water systems the ratio grows as $\mathcal{O}(N)$ with respect to the number of atoms N, with a Pearson coefficient equal to 0.995. In the protein systems the ratio still grows linearly but with a smaller slope: roughly a factor 2 is preserved.

Table 1 Global peak performance in percentage (%) assessed over a 50 femtoseconds MD trajectory. The Quadro GV100 was chosen to be the reference device

System/model	ANI	AMOEBA	Hybrid
DHFR	19.42	9.08	5.16
COX	28.13	10.52	n/a

To further analyze the computational bottleneck of HDNNP models, we evaluated the contribution of each of the model's constituents to the overall execution time (Fig. S1 ESI†). For small systems more than 40% of the cost is due to the gradients and AEV computations. The Tinker-HP neighbor list is less than 5% of the cost, demonstrating the performance of the implementation. For larger systems, the computational cost is largely dominated by the gradient's computation (*i.e.*, more than 50%). Thus, ML potential's computational performances are now mainly limited by back-propagation and not by the environment vector (the latter mainly being the memory bottleneck). Accelerating the gradient's estimation will therefore be of utmost importance for future implementations. Deep-HP also provides a keyword to automatically use mixed precision within PyTorch. The automatic mixed precision is using a combination of half and single precision operations without a severe loss on the model's accuracy.

2.3.3 Multi-GPU performance and scalability of ANI models within tinker-HP. In the following, we assess and discuss the multi-node performance of Deep-HP. The Jean Zay HPE SGI 8600 GPU system holds numerous computing nodes accelerated by 4 interconnected Tesla V100 devices each. Ideally, a parallel algorithm associated with a certain amount of resources (N processors for instance), whose load is equally distributed across all resources, will exactly perform N times faster. Experimentally, an intermediate step, occupied with communications, affects the performance to a varying degree depending on the size and pattern of these communications in comparison with the amount of calculations. When the number of allocated resources increases, global synchronizations induced by collective communications significantly slow down the parallel execution and, therefore, impact the asymptotic behavior of the strong scalability. Communication patterns and speed are subsequently the principal obstacles to achieve an ideal scaling. In our case, the domain decomposition method coupled with ANI offers an up-bounded communication pattern, which allows the use of several nodes without enduring severe performance loss too quickly, as it is the case with multi-node PFF on GPUs.[17] As displayed in Fig. 1b, we are able to scale up to 11 nodes (44 devices) for an 864 000 atom water box, before suffering from communication overheads and insufficient load. On the other hand, note that an accurate estimation of the gradients for each atom requires a complete knowledge of its surrounding environment up to a predetermined distance. The current implementation is however not optimal for a large number of processes and the performance starts to cap when half the minimum length of a domain equals the cutoff distance of the atomic environments. This is due to some redundancy between processes for the calculation of AEVs and energies of atoms from neighbouring domains. To illustrate this effect, we made an estimation of the performance in the case of no computational redundancy and plotted it for every test case in dashed lines within Fig. 1b. As anticipated, dealing with this effect can offer a significant 40% boost in the parallel run as is

Table 2 Performance of the ANI-2X neural network in Deep-HP in terms of molecular dynamics simulation production (ns per day) for selected water boxes of increasing sizes using Nvidia V100 and A100 GPU cards[a]

Systems (number of atoms)/number of GPU devices	1	4	8	16	28	44	68	84	100	124
GPU V100										
Puddle (96 000)	0.11	0.27	0.44	0.67	0.70	0.78	0.91	1.05	1.05	1.05
Pond (288 000)	n/a	0.11	0.19	0.31	0.46	0.57	0.66	0.67	0.71	0.71
Lake (864 000)	n/a	n/a	0.07	0.10	0.19	0.26	0.33	0.40	0.48	0.40
Bay (2 592 000)	n/a	—	n/a	0.04	0.06	0.09	0.14	n/a	n/a	n/a
Sea (7 776 000)	n/a		—	—		n/a	0.04	0.05	0.06	0.06
GPU A100										
Puddle (96 000)	0.16	0.41	0.63	n/a			—			n/a
Pond (288 000)	n/a	0.16	0.26	n/a			—			n/a
Lake (864 000)	n/a	n/a	0.11	n/a			—			n/a
Theoretical performance (V100)										
Puddle (96 000)	0.11	0.27	0.46	0.75	0.79	0.90	1.14	1.39	1.40	1.40
Pond (288 000)	0.03	0.11	0.20	0.33	0.49	0.65	0.77	0.89	0.88	0.89
Lake (864 000)	0.01	0.02	0.07	0.14	0.21	0.30	0.38	0.49	0.59	0.49
Bay (2 592 000)	0.004	0.007	0.02	0.06	0.08	0.12	0.16	n/a	n/a	n/a
Sea (7 776 000)	0.001	0.003	0.005	0.009	0.02	0.03	0.06	0.08	0.10	0.10

[a] n/a: not available.

observed for the sea water box. Thus, future implementations should address this issue in order to maximize multi-node performance. The test machines we used were also not optimal and do not provide fast interconnect between nodes. The observed A100 50% boost coupled to improved node interconnections will certainly be extremely beneficial to Deep-HP (we could not get access to a large recent A100 cluster and were limited to a single DGX-A100 node). Nevertheless, the current implementation can already be considered as a game changer for ANI/ANI-2X DNN simulations as the use of several GPUs already provides the capability to produce ns per day molecular dynamics simulations on hundreds of thousands of atom systems (see detailed benchmarks in Table 2).

2.3.4 Accelerating hybrid simulations: multi-timestep integrators (RESPA/RESPA1) and reweighting strategies

2.3.4.1 Multi-timestep integrators (RESPA/RESPA1). As fast as the ANI model can be compared to Density Functional Theory (10^6 factor speedup), ANI remains far more computationally demanding than polarizable force fields (see the ESI, Tables S1 and S2†) and the stiff intramolecular interactions reproduced by the MLP limits the integration time-step to "*ab initio*" 0.2–0.3 fs values, thus making the study of large proteins on long biological timescales a daunting task. One way to speed up MD is to use larger time steps through multi-time-stepping (MTS) methods thanks to a hybrid model. As discussed in Section 2.5, we decided to introduce the ANI-2X/AMOEBA model, that is, coupling a very accurate MLP for small molecules (ANI) to a PFFS designed to produce accurate condensed phase simulations of solvated proteins (AMOEBA). Typical MTS schemes exploit the separability of the potential energy into a computationally expensive, slowly varying part and a cheap, quickly varying part, and use a specific integration scheme, RESPA,[63] that allows for less frequent evaluations of the expensive part. In particular, in the context of the AMOEBA PFF, Tinker-HP uses either a bonded/non-bonded splitting or a three-stage separation between bonded, short-range non-bonded and long-range non-bonded interactions[64] (denoted as RESPA1 in the rest of the text). In both cases, temperature control is made through a BAOAB discretization of a Langevin equation.[65] In this context, the bonded forces are integrated using a small 0.2–0.3 fs time-step and the outermost time-step can be taken as 2 fs or 6 fs depending on the splitting. These can be further pushed by using Hydrogen Mass Repartitioning (HMR).[64,66] These integration schemes extend the applicability of PFFs to a longer time-scale reducing the gap with classical FFs, as demonstrated with recent simulations of tens of μs of the SARS-CoV2 M^{pro} protease.[2]

Even though MLPs are much less expensive than *ab initio* calculations, the most common MLPs with feed-forward neural networks remain more computationally demanding than FFs, even polarizable ones (see the ESI, Table S1†). To reduce this gap, towards simulating large biological systems, we combined our hybrid ANI-2X/AMOEBA model to MTS integrators using the RESPA scheme. We assume that AMOEBA is a good approximation of the ML potential for the isolated solute so that their energy difference $\Delta V_{ML}(P) = V_{ML}(P) - V_{AMOEBA}(P)$ should produce small forces that can be integrated using a larger time-step. This is done in the same spirit as Liberatore *et al.*[67] that studied such an integration scheme in the context of accelerating *ab initio* molecular dynamics. We thus associate this difference with the non-bonded part of the AMOEBA model and end up with the following separation:

$$V_{HYB}^{fast}(P \cup W) = V_{AMOEBA}^{bond}(P \cup W) \quad (8)$$

$$V_{HYB}^{slow}(P \cup W) = \Delta V_{ML}(P) + V_{AMOEBA}^{nonbond}(P \cup W) \quad (9)$$

where V_{HYB}^{fast} is evaluated every inner time-step and V_{HYB}^{slow} every outer one. In the RESPA1 framework, the potential energy difference $\Delta V_{ML}(P)$ is associated with the long-range interactions and evaluated at the outermost time-step.

To assess the accuracy of each integrator we computed the solvation free energy of two solutes with the hybrid model described above: the benzene molecule solvated in a cubic box of 996 water molecules with a 31 Å edge and a water molecule in a cubic box of 3999 other water molecules with a 49 Å edge. For each of these systems and integrators, we computed their solvation free energy by running 21 independent trajectories of 2 ns and 5 ns where the ligand is progressively decoupled from its water environment, first by annihilating its permanent multipoles and polarizabilities and then by scaling the associated van der Waals interactions (while using a softcore). The trajectories were run in the *NPT* ensemble at 300 K and 1 atmosphere using a Berendsen barostat and either a Bussi thermostat[68] (when Velocity Verlet is used) or a Langevin one for the MTS simulations as mentioned previously. The free energy differences were then computed using the BAR method.[69,70] Results were compared with a reference Velocity-Verlet integrator using a 0.2 fs time-step. The AMOEBA bonded forces were always evaluated every 0.25 fs. In the case of a bonded/non-bonded split, the non-bonded forces were evaluated either every 1 or 2 fs, and in the case where the non-bonded forces are further split between short-range and long-range ones, the short-range non-bonded forces were evaluated every 2 fs and the long-range ones either every 4 fs or 6 fs. As explained above, the MLP forces are always computed at the outermost time-step.

The accuracy of the results is displayed in Table 3. RESPA1 approaches, despite being operational, appear more sensitive to the system and do not always lead to the desired result in terms of free energies and should be restricted to simple simulation purposes. Therefore, the tighter RESPA (0.25/1 and 0.25/2) integrators are found to be good compromises between accuracy and computational gain. Table 4 shows the speedup of the hybrid model with various MTS setups compared to reference Velocity Verlet ANI-2X/AMOEBA simulation with a 0.2 fs time-step and Velocity Verlet ANI simulations with a 0.2 fs. In practice, speedups are system-dependent, but RESPA techniques always lead to a consequent acceleration compared with the tighter accuracy integration scheme (Verlet) for an ANI solute in a polarizable AMOEBA solvent and compared to pure ANI (Verlet 0.2 fs) simulations. These integrators thus extend the applicability of machine learning-driven molecular dynamics to larger biologically relevant systems and to longer-time-scale simulations. In practice, the resulting performance gain helps to reduce the computational gap between ANI and AMOEBA that is initially about more than a factor 30 (see the ESI, Table S1†).

Table 4 Relative speedup of hybrid models with RESPA (R) and RESPA1 (R1) integrators calculated with respect

splits	0.2	0.25/1	0.25/2	0.25/2/4	0.25/2/6
Benzene[a]	1.0	4.74	8.42	14.51	18.17
Water[a]	1.0	4.39	8.07	12.58	—
Benzene[b]	1.21	5.74	10.20	17.57	22.00
Water[b]	2.03	8.92	16.40	25.57	—
Integrator-type	V	R	R	R1(HMR)	R1(HMR)

[a] Hybrid model Velocity-Verlet (V) 0.2 fs time step. [b] ANI only with Velocity-Verlet (V) 0.2 fs time step.

2.3.4.2 Accelerating hybrid simulations: an alternative reweighting strategy. Concerning the proposed multi-timestep approach, it is important to note that since we assume that AMOEBA is a good approximation of the ML potential for the isolated solute, the present acceleration strategy is not possible when this condition is not fulfilled. In practice, it could happen in the event of an intramolecular reaction within the DNN solute. Indeed, ANI-2X being a reactive potential, it is sometimes able to produce intramolecular proton transfers in some specific cases, *i.e.*, when donor and acceptor functional groups are present. In contrast, AMOEBA is a non-reactive force field that will always stay in its initial electronic state. Therefore, an intra-ligand chemical reaction would desynchronize the two potentials and therefore stop the simulation. In the rare case of such an event, it is always possible to use a two-step approach and to produce the BAR simulation windows thanks to fast AMOEBA, non-reactive, trajectories. Then one can analyse the AMOEBA snapshots by computing the corresponding ANI-2X/AMOEBA energies to correct the AMOEBA free energy evaluation using a rigorous BAR reweighting[70,71] (details can be found in the ESI,† see Section 2.2). Such an alternative approach preserves the advantage of speed since the computation of the costly DNN gradients is avoided.

3 Results

3.1 Solvation free energies

3.1.1 Computational details. To assess further the performance of the ANI-2X/AMOEBA hybrid model, we extended our solvation free energy tests to a variety of small molecules under

Table 3 Solvation free energy (kcal mol^{-1}) comparison for the benzene and water molecules. Comparison between experimental, AMOEBA and hybrid ANI-2X/AMOEBA results using Velocity Verlet, BAOAB-RESPA and BAOAB-RESPA1 integrators. H corresponds to the use of hydrogen mass repartitioning (HMR). Simulations were performed in the *NPT* ensemble with 2 ns and 5 ns (in parentheses) BAR windows, with the BAOAB-RESPA/RESPA1 integrators

	Exp.	AMOEBA	V (0.2)	R (0.25/1)	R (0.25/2)	R1^{H} (0.25/2/4)	R1^{H} (0.25/2/6)
Benzene	−0.87	−0.37	−0.83	−0.97 (−0.90)	−0.87 (−0.88)	−1.69 (−1.69)	−1.60
Water	−6.32	−5.62	−6.33	−6.29 (−6.23)	−6.21 (−6.22)	−6.39 (−6.33)	—

both aqueous and non-aqueous conditions, as described in ref. 72 and [73]. The solvents considered, along with their dielectric permittivity values, are as follows: toluene ($\varepsilon = 2.38$), acetonitrile ($\varepsilon = 36.64$), DMSO ($\varepsilon = 47.24$) and water ($\varepsilon = 77.16$). Further details regarding the solutes can be found in the ESI.†

We withdrew molecules from the dataset that contained chemical elements not available in ANI-2X, resulting in a total of 38 molecules solvated in water (taken from ref. 43), 20 molecules solvated in toluene, 6 in acetonitrile and 6 in DMSO (taken from Essex *et al.*).[72] All the systems were prepared following the standard equilibration protocol: after a geometry optimization, they were progressively heated up to 300 K in *NVT* and then equilibrated for 1 ns in the *NPT* ensemble at the same temperature and 1 atmosphere. In all cases, we used the most simple multiple time-step integrator presented above with a 0.25 fs time-step for bonded terms and 1 fs for the outermost one. The Bussi thermostat and the Berendsen barostat were used. The van der Waals interaction cutoff was chosen at 12 Å and the electrostatic interactions were handled with the Smooth Particle Mesh Ewald method[44] with a 7 Å real space cutoff and default Tinker-HP grid size. We used the same scheme as before to decouple the systems from their environment with 21 independent windows of 2 ns. For solvation free energies in water we also pushed the ANI-2X/AMOEBA simulation windows up to 5 ns. Water as a solvent has been intensively studied as it constitutes a core component driving drug design and as it allows testing for the validity of various computational methods and models.[4,74] The results are compared with experimental data and with the AMOEBA ones. ANI-2X and AMOEBA standard parametrizations[72,73] were used.

3.1.2 Results and discussion. The experimental, AMOEBA and ANI-2X/AMOEBA solvation free energy data are provided in Fig. 2 and Tables S3–S7 of the ESI.† We start with the most challenging solvent, *i.e.* water, which is highly polar and known to be difficult for neural networks. In order to match the recently published AMOEBA Poltype2 (ref. 43) study, we first performed trajectories of 5 ns (instead of 2 ns for other solvents). While we kept most of the poltype2 AMOEBA

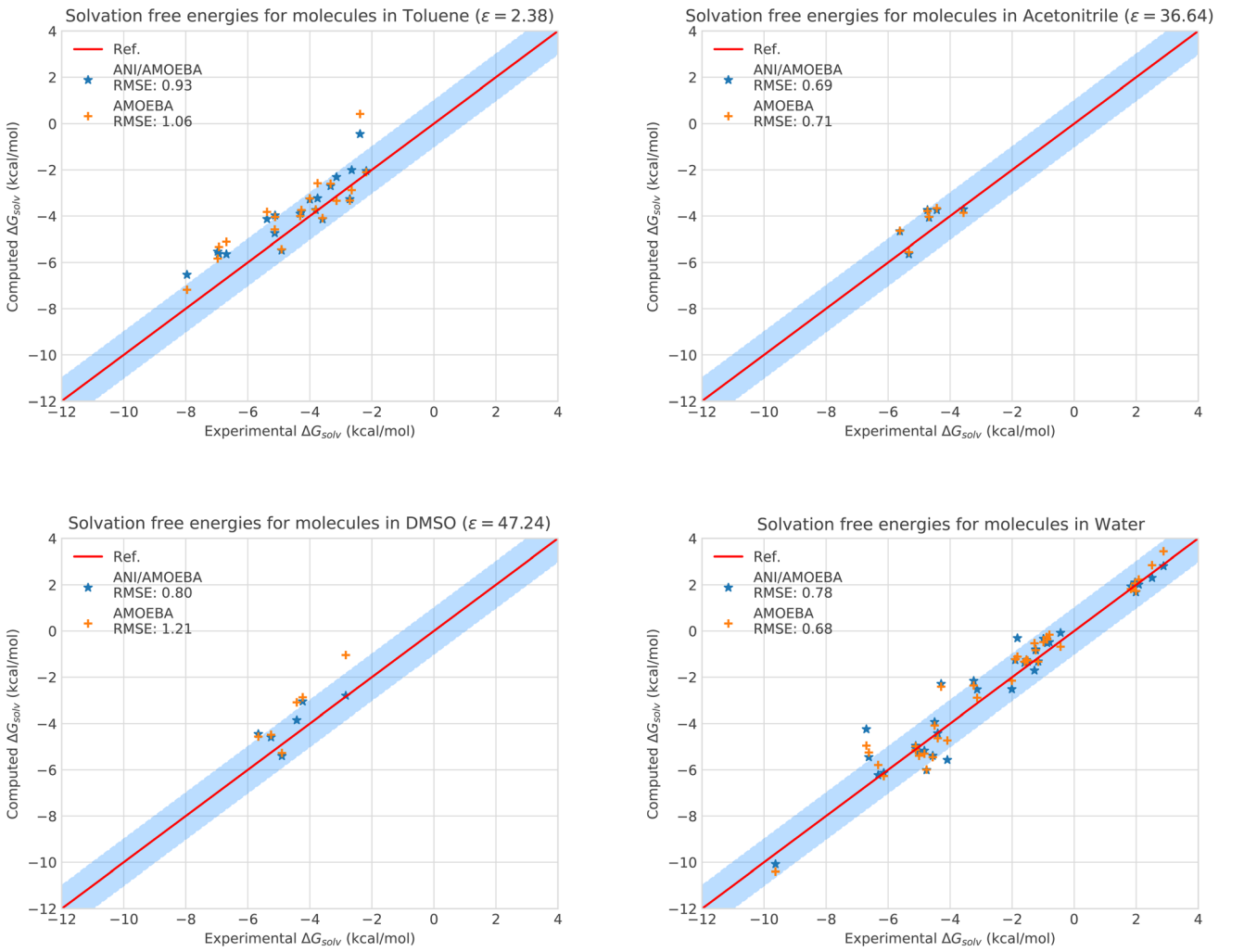


Fig. 2 Solvation free energies of molecules in different solvents computed with AMOEBA (orange) from ref. 72 and 73 *versus* hybrid model ANI-2X/AMOEBA (blue) and experiment (red). The blue domain corresponds to the so-called chemical accuracy: error of 1 kcal mol^{-1} w.r.t. experiment.

parameters unchanged, we reparametrized the water, phenol, methylamine and dimethylamine ligands (denoted by R in Table S3†) with the latest version of the Poltype2 software as they were notably performing below the usual AMOEBA standards. Overall, despite the difficult polar solvent, ANI-2X/AMOEBA performs extremely well compared to AMOEBA, exhibiting an RMSE of 0.78 kcal mol^{-1} *vs.* 0.68 kcal mol^{-1} for the polarizable force field. This is a very good performance for ANI-2X/AMOEBA since the AMOEBA water model is well-known for its accuracy and capabilities to reproduce numerous water-related experimental data.[38] To assess the statistical error on solvation free energies, we performed another full run of ANI-2X/AMOEBA (see the ESI, Tables S11 and S12†). The averaged statistical uncertainty amounts for 0.17 kcal mol^{-1} which is consistent with the AMOEBA literature which usually reports errors in the 0.15–0.25 kcal mol^{-1} range for solvation studies.[75,76] We also investigated the BAR source of error (*via* bootstrapping[76]) which amounts for 0.04 kcal mol^{-1}. If a full assessment of statistical errors, *i.e.*, involving multiple simulation replicas is currently out of reach of our computational capabilities due to the use of neural networks, it is nevertheless possible to conclude that ANI-2X/AMOEBA and AMOEBA yield comparable results in water. It is a remarkable result for ANI-2X/AMOEBA that highlights the high accuracy of ANI-2X.

Of course, since water is particularly challenging, we anticipate that ANI-2X would exhibit a gain in accuracy when dealing with apolar solvents. This is clearly the case. For example, for toluene which is a less polar solvent (see Table S5†), the hybrid ANI-2X/AMOEBA results tend to be more accurate than the AMOEBA ones (while staying in the statistical uncertainty), with a respective RMSE of 0.93 kcal mol^{-1} *vs.* 1.06 kcal mol^{-1} for AMOEBA. In acetonitrile, ANI-2X/AMOEBA is equivalent to AMOEBA (0.69 kcal mol^{-1} *vs.* 0.71 kcal mol^{-1}). However, in DMSO, ANI-2X/AMOEBA performs significantly better than AMOEBA, with a respective RMSE of 0.80 kcal mol^{-1} *vs.* 1.21 kcal mol^{-1} for AMOEBA. Thus, ANI-2X/AMOEBA and AMOEBA results are within the statistical error for 3 of the studied solvents (including water) while ANI-2X/AMOEBA performs better for DMSO highlighting the high accuracy of ANI-2X. These data confirm the robustness of ANI-2X/AMOEBA in a difficult polar solvent like water once long-range and many-body effects are present. A grasp of its applications will be briefly discussed in the section dedicated to host–guest systems. On the technical point of view, the RESPA acceleration strategy has also been shown to be particularly effective for this solvation study.

In the next section, we go a step further in terms of complexity and report the hybrid model performance on 14 challenging host–guest systems taken from the SAMPL competitions.[77,78]

3.2 Host–guest binding free energies: SAMPL challenges

3.2.1 Computational details. This section is dedicated to the measure of the accuracy of the ANI-2X/AMOEBA framework compared to AMOEBA for evaluating host–guest binding free energies. Indeed, AMOEBA is known as one of the most accurate approaches for such studies (see the discussion around the SAMPL challenge[79]) and reaching such accuracy would be a landmark for hybrid neural network simulations. We considered the absolute binding free energy values of 13 guests from the 14 SAMPL4 CB[7]–guest challenge.[78] We will consider separately the C5 compound that was previously shown[78] to be a specific outlayer case. We completed the study adding a fourteen complex, the G9 guest taken from the SAMPL6 cucurbit[8]uril host–guest challenge. Free energies were calculated with the hybrid ANI-2X/AMOEBA model as the difference between the free energy of decoupling the ligands within the host and in solution. The optimized structures and parameters for the AMOEBA FF were taken from the literature.[75,78,80,81] Again, in order to evaluate the impact of the ANI-2X contributions, no AMOEBA specific parametrization has been performed. These ligands are challenging as they are charged, flexible and large, usually leading to difficulties in the prediction of binding free energies.[78] The same protocol (2 ns windows) as before was used except that the RESPA outer time-step was changed from 1 fs to 2 fs which still gives a satisfactory accuracy, see Table 3. We also provide the free energy values for extended simulations with 5 ns windows in order to explore the accuracy convergence.

3.2.2 Results and discussion. The binding free energies of the host–guest systems are depicted in Fig. 3 and in Tables S8–S10.† Let's focus first on the accuracy of the ANI-2X/AMOEBA prediction. Overall, the hybrid potential results perform better than the available AMOEBA data reaching an accuracy in the range of chemical accuracy, *i.e.*, 1 kcal mol^{-1} average error w.r.t. experiment. ANI-2X/AMOEBA gives an RMSE of 0.94 kcal mol^{-1} *versus* 1.81 kcal mol^{-1} for AMOEBA. It is important to note that in this very challenging testset, all the ligands are charged and encompass a net charge of 1 or 2. As for solvation free energies, the combination of the ANI-2X ligands with the polarizable AMOEBA solvent, host and long-range effects appears to be a powerful tool. Due to computational limitations because of

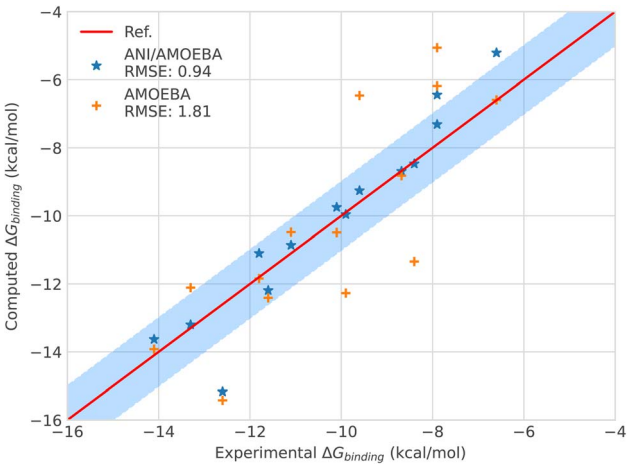


Fig. 3 Binding free energies of host–guest systems of the SAMPL4 and SAMPL6 blind challenges with AMOEBA (orange) from ref. 78 *versus* hybrid model ANI-2X/AMOEBA (blue) and experimental (red). The blue domain corresponds to the so-called chemical accuracy: error of 1 kcal mol^{-1} w.r.t. experiment.

the extensive use of neural networks, we did not resort to extensive statistical error analysis but it is clear that despite the fact that binding free energy uncertainties are usually roughly twice larger than those obtained for solvation studies, ANI-2X/AMOEBA results exhibit a significant improvement over AMOEBA (see a detailed discussion about the statistical uncertainties that one could expect for such studies in ref. 75 and [76]). If we go more in detail, compound by compound, ANI-2X/AMOEBA exhibits a larger error than AMOEBA for only the C13, C8 and C3 guest ligands. For C13, predictions are both within 0.5 kcal mol^{-1} from experiment. For C8, ANI-2X/AMOEBA also stays within 1 kcal mol^{-1} of error (0.7 kcal mol^{-1}). C8 has been shown to be associated with high enthalpy changes throughout binding[78] and such a change can be traced back to some gains in terms of H-bond interactions from the solution to the host–guest complex. It suggests that improvements of ANI-2X towards improved H-bond treatment could be beneficial. This is consistent with our findings on the solvation free energies where AMOEBA performs slightly better than ANI-2X. Concerning C3, the case is more complex and we review our results below in the same section in link with the discussion on integrators' performances. Only two compound predictions did not reach chemical accuracy: C9 and C10. However, in these cases, the initial AMOEBA error is improved (divided by 2 for C10) using ANI-2X/AMOEBA confirming the higher accuracy of the hybrid model. This result could be associated with slow sampling convergence as noticed by Ren *et al.*[78] It is worth reporting that in the case of the last compound, *i.e.* the SAMPL6 host–guest system, the ANI-2X/AMOEBA results almost exactly match the experimental results (see the ESI, Table S8†). Finally, we also present in the ESI (Table S10†), the results for the C5 compound that was removed from the testset. These results confirm the initial assessment by Ren *et al.*[78] and would require further investigation (protonation states, binding modes, sampling time *etc.*…) going beyond the scope of the present work.

Looking in detail at the free energy acceleration strategy, we were overall able to use a RESPA approach on 12 of the 15 (14 + C5) tested ligands. The integrator was not stable enough for the C2, C3 and C4 compounds (see Fig. 3 and ESI, Table S9†). This is due to different reasons. First, C2 and C4 exhibited notably higher differences between the ANI-2X and AMOEBA potentials compared to other ligands. This can be easily understood when considering that C2 and C4 are actually associated with the two largest AMOEBA dataset deviations from the experimental reference values (errors of 3.14 and 2.94 kcal mol^{-1}, for C2 and C4 respectively). Since our initial choice was to not perform any specific AMOEBA re-parametrization or ANI-2X dataset modification, the strategy required to either use a tighter, but computationally inefficient Verlet/0.2 fs integration or to perform an ANI-2X/AMOEBA BAR reweighting of a non-reactive AMOEBA set of trajectories, as discussed at the end of Section 2.3.4. Due to the computational constraints, we chose the reweighting strategy that benefits from the efficiency of Tinker-HP to generate AMOEBA trajectories. Table S9 (ESI†) displays the ANI-2X/AMOEBA results obtained for C2 and C4. They are found to be in very good agreement with experiment with errors of 0.34 and 0.07 kcal mol^{-1} respectively. Again, the hybrid potential notably outperforms AMOEBA in these cases as ANI-2X clearly helps to improve the accuracy for these two compounds. For the last ligand, C3, the nature of the problem appeared to be very different as the AMOEBA free energy prediction was almost perfect compared to experiment. In fact, we do not have a parametrization issue here and C3 represents the only case where a reactivity event occurred within our simulations. Indeed, when binding to the host, the C3 ligand adopts a cyclic conformation where its terminal OH and NH_3 groups strongly interact. This is well captured by AMOEBA. Due to its reactive nature, the ANI-2X potential is able to produce MD trajectories that include proton transfers between the groups suggesting that, for ANI-2X, the compound is actually a mix of two electronic states. As discussed in Section 2.3.4, this situation is simply incompatible with a hybrid RESPA strategy. Again, we performed an ANI-2X/AMOEBA BAR reweighting computation using the well-defined initial AMOEBA electronic state to produce non-reactive classical trajectories. This led to a result apparently less in line with experiment than the AMOEBA one (1.76 kcal mol^{-1} *vs.* 0.01 kcal mol^{-1} for AMOEBA) which was anticipated as ANI-2X tends to disfavor the initial state. A solution would be to compute all possible states explored by ANI-2X/AMOEBA. Indeed, many things remain to be solved in the modeling of the SAMPL4 dataset. For example, in the SAMPL4 challenge overview, Muddana *et al.*[81] reviewed the experimental conditions and concluded that it could be important to take into account the salt conditions and to go beyond the simple box neutralization. Indeed, in the event of a proton transfer, a new ionic species being created, it would be interesting to study its interaction with different solutions of increasing ionic strength, especially in our case where the full simulation includes polarization effects. We have not done it at this stage as it would require a large number of additional simulations and we decided to retain the present C3 free energy prediction that could probably be improved in a forthcoming study. In any case, with C3, ANI-2X brings additional interpretative insights on the nature of the ligand. In the near future, it will also be interesting to investigate further the reactivity capabilities of the ANI-2X/AMOEBA approach. Finally, it is worth noting that C3 is the weakest binder of the series. ANI-2X/AMOEBA still predicts it as such in terms of the relative free energy of binding compared to the other compounds.

Overall, the hybrid ANI-2X/AMOEBA model results are in good agreement with experimental results, reaching, as for the solvation free energy studies, an accuracy in the range of chemical accuracy (average error of 0.94 kcal mol^{-1} *vs.* experiment on the dataset) and dividing the initial AMOEBA error by 2. ANI-2X/AMOEBA can accurately predict binding free energies of flexible charged systems and the simulations clearly benefit from the addition of ANI-2X. Finally, in contrast with the results obtained by Lahey and Rowley[57] that showed the difficulties of the ANI-2X potential for modeling charged systems within a hybrid embedding approach with non-polarizable force fields, we observed accurate results even for charged systems. This is due to a combination of factors linked to many-body and long-range effects and to solvation. Indeed, in the ANI-2X/AMOEBA

framework, the charged ligands are embedded in a flexible polarizable solvent that can adapt its dipolar moment to its micro-environment net charges (see ref. 3 and 82 for discussions), providing extra flexibility for the hybrid polarizable embedding approach. For example, the hybrid approach yields good results for nitro-methane, which is globally neutral but still bears two charged groups.

4 Conclusion and perspectives

We first introduced Deep-HP, a novel massively parallel multi-GPU neural network platform which is a new component of the Tinker-HP molecular dynamics package. Deep-HP allows users to import their favorite Pytorch/TensorFlow Deep Neural Network models within Tinker-HP. While Deep-HP enables the simulation of millions of atoms thanks to its MPI/domain decomposition setup, it introduces the possibility of reaching ns routine production simulation for hundreds of thousands of atom biosystems with advanced neural network models such as ANI-2X. The platform capabilities have been demonstrated by simulating large biologically relevant systems on up to 124 GPUs with ANI-2X.

Since the platform allows the coupling of state-of-the-art polarizable force fields with any ML potential, we developed a new hybrid deep neural networks/polarizable potential that uses the ANI-2X ML potential for the solute–solute interactions and the AMOEBA polarizable force field for the rest. The development of the hybrid potential was motivated by the capability of AMOEBA to accurately model water–solute and water–water interactions, whereas a neural network such as ANI is better able to capture complex intramolecular interactions at an accuracy approaching the CCSDT(T) gold standard of computational chemistry.[54]

We extended our hybrid model computational capabilities by designing RESPA-like multi-timestep integrators that can speed up simulations up to more than an order of magnitude with respect to Velocity Verlet 0.2 fs. In that context, the relative speedup of AMOEBA compared to the hybrid ANI-2X/AMOEBA dropped from 40 to 2. The hybrid approach offers the inclusion of physically motivated long-range effects (electrostatics and many-body polarization) and the capability to perform efficient Particle Mesh Ewald periodic boundary condition simulations including polarizable counter ions. It also allows us to benefit from the capability of the ANI-2X neural network to accurately describe the ligand potential energy surface leading to high-resolution exploration of its conformational space through the hybrid model MD simulation. The combination of these approaches allows us to treat any type of ligands, including charged ones and opens the door to routine long timescale simulations using NNPs/PFFs up to million-atom biological systems, offering considerable speedup compared to traditional ligand binding QM/MM simulations.

Our hybrid model accuracy was first assessed on solvation free energies of 70 molecules, with a large panel of different functional groups including charged ones, within three non-aqueous solvents and water. The hybrid model is shown to perform well, reaching similar or better accuracy compared to the AMOEBA polarizable force field. Such results open a path towards the simulation of complex biological processes with neural networks for which the environment polarizability is important.[3,4,82] We then reported the performance of our hybrid model on the binding free energies of 14 host–guest challenging systems taken from the SAMPL host–guest binding competitions. Although most of the ligands are charged, our hybrid model is able to reach performances superior to those of AMOEBA despite the complex chemical environments. Overall, ANI-2X/AMOEBA is shown to reach an accuracy in the range of the chemical accuracy (average errors < 1 kcal mol^{-1} w.r.t. experiment) on the testsets for both solvation and absolute binding free energies. Further work is required to assess the statistical uncertainties linked to such hybrid simulations but the advances in software and HPC will certainly enable such an assessment in the incoming years. Of course, it is important to note that, in some cases, AMOEBA alone is able to reach sub-kcal mol^{-1} accuracy (see for example the SAMPL 8 results).[79] However, it is not always the case (see SAMPL 6 and 7 results)[75,76] and seeing a hybrid neural network technology reaching such an accuracy limit is clearly a new step forward.

ANI-2X also provides new features such as the possibility to detect chemical modifications of the ligand thanks to the neural network reactive nature. As the model improves, it could be an important asset for such simulations. As discussed, an accurate AMOEBA parametrization is important and it will be interesting to systematically better converge the level of parametrization of AMOEBA and ANI-2X ligands in order to benefit from maximal multi-timestep acceleration. This should be easily achievable thanks to the recent improvements of the Poltype2 AMOEBA automatic parametrization framework.[43] In this line, adaptive-timestep alternatives to multi-timestepping using Velocity Jumps[83] would also be beneficial and are under investigation. These reactivity events also led us to introduce an accurate reweighting strategy. Since it is computationally efficient and avoids the costly computation of DNN gradients, it may become one of the strategies for free energy predictions. Further work will analyse the multiple possibilities of neural network reweighting setups in order to assess their computational efficiency.

Overall, the Deep-HP platform, which takes advantage of state-of-the-art Tinker-HP GPU code, was able to produce within a few days more than 10 μs of hybrid NNPS/PFFs molecular dynamics simulations which is, to our knowledge, the longest MD biomolecular study encompassing neural networks performed to date. Such performances should continue to improve thanks to further Deep-HP optimizations, TorchANI updates and GPU hardware evolutions. Deep-HP will enable the implementation of the next generation of improved MLPs[84–86] and has been designed to be a place for their further development. It will include direct neural network coupling with physics-driven contributions going beyond multipolar electrostatics and polarization through the inclusion of many-body dispersion models.[87,88] As Deep-HP's purpose is to push a trained ML/hybrid model towards large scale production simulations, we expect extensions of the present simulation capabilities to other class of systems towards materials and catalysis applications. Overall, Deep-HP allows the present ANI-2X/AMOEBA hybrid

model to go a step further towards one of the grails of computation chemistry which is the unification within a reactive molecular dynamics many-body interaction potential of the short-range quantum mechanical accuracy and of long-range classical effects, at force field computational cost.

Data availability

Deep-HP is part of the Tinker-HP package which is freely accessible to Academics *via* GitHub: **https://github.com/TinkerTools/tinker-hp**. We are also providing a tutorial: **https://github.com/TinkerTools/tinker-hp/blob/master/GPU/Deep-HP.md**.

Author contributions

T. J. I., O. A. and T. P. performed simulations; O. A., O. I., T. J. I., L. L. and T. P. contributed the new code; L. L., O. I., T. J. I., P. R., T. P., and J.-P. P. contributed the new methodology; T. J. I., L. L., P. R., and J.-P. P. contributed the analytical tool; T. J. I., L. L., O. I., P. R., H. G., and J.-P. P. analyzed the data. T. J. I., L. L., T. P., H. G., O. I. and J.-P. P. wrote the paper; J.-P. P. designed the research.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement no. 810367), project EMC2 (JPP). Simulations have been performed at GENCI on the Jean Zay machine (IDRIS, Orsay, France) on grant no. A0070707671 and at TGCC (Bruyères le Châtel, France) on the Irène Joliot Curie machine. The work performed by H. G. and O. I. (PI) was made possible by the Office of Naval Research (ONR) through support provided by the Energetic Materials Program (MURI grant no. N00014-21-1-2476). This research is part of the Frontera computing project at the Texas Advanced Computing Center. Frontera is made possible by the National Science Foundation award OAC-1818253.

Notes and references

1 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, *Nature*, 2021, **596**, 583–589.

2 T. Jaffrelot Inizan, F. Célerse, O. Adjoua, D. El Ahdab, L.-H. Jolly, C. Liu, P. Ren, M. Montes, N. Lagarde, L. Lagardère, P. Monmarché and J.-P. Piquemal, *Chem. Sci.*, 2021, **12**, 4889–4907.

3 D. El Ahdab, L. Lagardère, T. J. Inizan, F. Célerse, C. Liu, O. Adjoua, L.-H. Jolly, N. Gresh, Z. Hobaika, P. Ren, R. G. Maroun and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2021, **12**, 6218–6226.

4 L. El Khoury, Z. Jing, A. Cuzzolin, A. Deplano, D. Loco, B. Sattarov, F. Hédin, S. Wendeborn, C. Ho, D. El Ahdab, T. Jaffrelot Inizan, M. Sturlese, A. Sosic, M. Volpiana, A. Lugato, M. Barone, B. Gatto, M. L. Macchia, M. Bellanda, R. Battistutta, C. Salata, I. Kondratov, R. Iminov, A. Khairulin, Y. Mykhalonok, A. Pochepko, V. Chashka-Ratushnyi, I. Kos, S. Moro, M. Montes, P. Ren, J. W. Ponder, L. Lagardère, J.-P. Piquemal and D. Sabbadin, *Chem. Sci.*, 2022, **13**, 3674–3687.

5 J. C. Phillips, D. J. Hardy, J. D. C. Maia, J. E. Stone, J. V. Ribeiro, R. C. Bernardi, R. Buch, G. Fiorin, J. Hénin, W. Jiang, R. McGreevy, M. C. R. Melo, B. K. Radak, R. D. Skeel, A. Singharoy, Y. Wang, B. Roux, A. Aksimentiev, Z. Luthey-Schulten, L. V. Kalé, K. Schulten, C. Chipot and E. Tajkhorshid, *J. Chem. Phys.*, 2020, **153**, 044130.

6 D. E. Shaw, J. P. Grossman, J. A. Bank, B. Batson, J. A. Butts, J. C. Chao, M. M. Deneroff, R. O. Dror, A. Even, C. H. Fenton, A. Forte, J. Gagliardo, G. Gill, B. Greskamp, C. R. Ho, D. J. Ierardi, L. Iserovich, J. S. Kuskin, R. H. Larson, T. Layman, L. Lee, A. K. Lerer, C. Li, D. Killebrew, K. M. Mackenzie, S. Y. Mok, M. A. Moraes, R. Mueller, L. J. Nociolo, J. L. Peticolas, T. Quan, D. Ramot, J. K. Salmon, D. P. Scarpazza, U. B. Schafer, N. Siddique, C. W. Snyder, J. Spengler, P. T. P. Tang, M. Theobald, H. Toma, B. Towles, B. Vitale, S. C. Wang and C. Young, *SC '14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2014, pp. 41–53.

7 M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess and E. Lindahl, *SoftwareX*, 2015, **1–2**, 19–25.

8 C. Kobayashi, J. Jung, Y. Matsunaga, T. Mori, T. Ando, K. Tamura, M. Kamiya and Y. Sugita, *J. Comput. Chem.*, 2017, **38**, 2193–2206.

9 J. W. Ponder and D. A. Case, *Protein Simulations*, Academic Press, 2003, vol. 66, pp. 27–85.

10 L. Monticelli and D. P. Tieleman, in *Force Fields for Classical Molecular Dynamics*, ed. L. Monticelli and E. Salonen, Humana Press, Totowa, NJ, 2013, pp. 197–213.

11 N. Gresh, G. A. Cisneros, T. A. Darden and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2007, **3**, 1960–1986.

12 J. Melcr and J.-P. Piquemal, *Front. Mol. Biosci.*, 2019, **6**, 143.

13 Y. Shi, P. Ren, M. Schnieders and J.-P. Piquemal, in *Polarizable Force Fields for Biomolecular Modeling*, John Wiley and Sons, Ltd, 2015, ch. 2, pp. 51–86.

14 Z. Jing, C. Liu, S. Y. Cheng, R. Qi, B. D. Walker, J.-P. Piquemal and P. Ren, *Annu. Rev. Biophys.*, 2019, **48**, 371–394.

15 L. Lagardère, L.-H. Jolly, F. Lipparini, F. Aviat, B. Stamm, Z. F. Jing, M. Harger, H. Torabifard, G. A. Cisneros, M. J. Schnieders, N. Gresh, Y. Maday, P. Y. Ren, J. W. Ponder and J.-P. Piquemal, *Chem. Sci.*, 2018, **9**, 956–972.

16 J. Huang, P. E. M. Lopes, B. Roux and A. D. MacKerell, *J. Phys. Chem. Lett.*, 2014, **5**, 3144–3150.
17 O. Adjoua, L. Lagardère, L.-H. Jolly, A. Durocher, T. Very, I. Dupays, Z. Wang, T. J. Inizan, F. Célerse, P. Ren, J. W. Ponder and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2021, **17**, 2034–2053.
18 A. P. Bartók, M. C. Payne, R. Kondor and G. Csányi, *Phys. Rev. Lett.*, 2010, **104**, 136403.
19 A. Thompson, L. Swiler, C. Trott, S. Foiles and G. Tucker, *J. Comput. Phys.*, 2015, **285**, 316–330.
20 V. Vovk, in *Kernel Ridge Regression*, ed. B. Schölkopf, Z. Luo and V. Vovk, Springer, Berlin, Heidelberg, 2013, pp. 105–116.
21 H. E. Sauceda, S. Chmiela, I. Poltavsky, K.-R. Müller and A. Tkatchenko, *J. Chem. Phys.*, 2019, **150**, 114102.
22 S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt and K.-R. Müller, *Sci. Adv.*, 2017, **3**, e1603015.
23 S. Chmiela, H. E. Sauceda, K.-R. Müller and A. Tkatchenko, *Nat. Commun.*, 2018, **9**, 3887.
24 H. E. Sauceda, S. Chmiela, I. Poltavsky, K.-R. Müller and A. Tkatchenko, *J. Chem. Phys.*, 2019, **150**, 114102.
25 O. Ivanciuc, in *Applications of Support Vector Machines in Chemistry*, John Wiley and Sons, Ltd, 2007, ch. 6, pp. 291–400.
26 A. P. Bartók, R. Kondor and G. Csányi, *Phys. Rev. B*, 2013, **87**, 184115.
27 A. Fabrizio, K. R. Briling and C. Corminboeuf, *Digital Discovery*, 2022, **1**, 286–294.
28 J. Behler and M. Parrinello, *Phys. Rev. Lett.*, 2007, **98**, 146401.
29 M. Gastegger, L. Schwiedrzik, M. Bittermann, F. Berzsenyi and P. Marquetand, *J. Chem. Phys.*, 2018, **148**, 241709.
30 J. Behler, *Chem. Rev.*, 2021, **121**, 10037–10072.
31 J. S. Smith, O. Isayev and A. E. Roitberg, *Chem. Sci.*, 2017, **8**, 3192–3203.
32 B. Lier, P. Poliak, P. Marquetand, J. Westermayr and C. Oostenbrink, *J. Phys. Chem. Lett.*, 2022, **13**, 3812–3818.
33 W. Jia, H. Wang, M. Chen, D. Lu, L. Lin, R. Car, E. Weinan and L. Zhang, *SC20: International conference for high performance computing, networking, storage and analysis*, 2020, pp. 1–14.
34 T. W. Ko, J. A. Finkler, S. Goedecker and J. Behler, *Nat. Commun.*, 2021, **12**, 398.
35 D. Loco, L. Lagardère, S. Caprasecca, F. Lipparini, B. Mennucci and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2017, **13**, 4025–4033.
36 D. Loco, L. Lagardère, O. Adjoua and J.-P. Piquemal, *Acc. Chem. Res.*, 2021, **54**, 2812–2822.
37 D. Loco, L. Lagardère, G. A. Cisneros, G. Scalmani, M. Frisch, F. Lipparini, B. Mennucci and J.-P. Piquemal, *Chem. Sci.*, 2019, **10**, 7200–7211.
38 P. Ren and J. W. Ponder, *J. Phys. Chem. B*, 2003, **107**, 5933–5947.
39 J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. A. DiStasio, M. Head-Gordon, G. N. I. Clark, M. E. Johnson and T. Head-Gordon, *J. Phys. Chem. B*, 2010, **114**, 2549–2564.
40 N. L. Allinger, Y. H. Yuh and J. H. Lii, *J. Am. Chem. Soc.*, 1989, **111**, 8551–8566.
41 B. T. Thole, *Chem. Phys.*, 1981, **59**, 341–350.
42 T. A. Halgren, *J. Am. Chem. Soc.*, 1992, **114**, 7827–7843.
43 B. Walker, C. Liu, E. Wait and P. Ren, *J. Comput. Chem.*, 2022, **43**, 1530–1542.
44 U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee and L. G. Pedersen, *J. Chem. Phys.*, 1995, **103**, 8577–8593.
45 L. Lagardère, F. Lipparini, E. Polack, B. Stamm, E. Cances, M. Schnieders, P. Ren, Y. Maday and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2015, **11**, 2589–2599.
46 J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. A. DiStasio Jr, et al., *J. Phys. Chem. B*, 2010, **114**, 2549–2564.
47 A. Grossfield, P. Ren and J. W. Ponder, *J. Am. Chem. Soc.*, 2003, **125**, 15671–15682.
48 J. C. Wu, J.-P. Piquemal, R. Chaudret, P. Reinhardt and P. Ren, *J. Chem. Theory Comput.*, 2010, **6**, 2059–2070.
49 Y. Shi, Z. Xia, J. Zhang, R. Best, C. Wu, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2013, **9**, 4046–4063.
50 C. Zhang, C. Lu, Z. Jing, C. Wu, J.-P. Piquemal, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2018, **14**, 2084–2108.
51 J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev and A. E. Roitberg, *J. Chem. Phys.*, 2018, **148**, 241733.
52 J. S. Smith, O. Isayev and A. E. Roitberg, *Sci. Data*, 2017, **4**, 170193.
53 J. S. Smith, R. Zubatyuk, B. Nebgen, N. Lubbers, K. Barros, A. E. Roitberg, O. Isayev and S. Tretiak, *Sci. Data*, 2020, **7**, 134.
54 C. Devereux, J. S. Smith, K. K. Huddleston, K. Barros, R. Zubatyuk, O. Isayev and A. E. Roitberg, *J. Chem. Theory Comput.*, 2020, **16**, 4192–4202.
55 L. Zhang, J. Han, H. Wang, R. Car and W. E, *Phys. Rev. Lett.*, 2018, **120**, 143001.
56 H. Wang, L. Zhang, J. Han and W. E, *Comput. Phys. Commun.*, 2018, **228**, 178–184.
57 S.-L. J. Lahey and C. N. Rowley, *Chem. Sci.*, 2020, **11**, 2362–2368.
58 J. Norberg and L. Nilsson, *Biophys. J.*, 2000, **79**, 1537–1553.
59 A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2019, vol. 32, pp. 8024–8035.
60 M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015, **https://www.tensorflow.org/**.

61 F. Chollet, et al., *Keras*, 2015, **https://github.com/fchollet/keras**.
62 A. D. MacKerell Jr, B. Brooks, C. L. Brooks III, L. Nilsson, B. Roux, Y. Won and M. Karplus, in *CHARMM: The Energy Function and Its Parameterization*, Wiley and Sons, 2002.
63 M. Tuckerman, B. Berne and G. Martyna, *J. Chem. Phys.*, 1992, **97**, 1990–2001.
64 L. Lagardère, F. Aviat and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2019, **10**, 2593–2599.
65 B. Leimkuhler and C. Matthews, *J. Chem. Phys.*, 2013, **138**, 05B601_1.
66 R. Zhou, E. Harder, H. Xu and B. J. Berne, *J. Chem. Phys.*, 2001, **115**, 2348–2358.
67 E. Liberatore, R. Meli and U. Rothlisberger, *J. Chem. Theory Comput.*, 2018, **14**, 2834–2842.
68 G. Bussi, D. Donadio and M. Parrinello, *J. Chem. Phys.*, 2007, **126**, 014101.
69 C. H. Bennett, *J. Comput. Phys.*, 1976, **22**, 245–268.
70 J. Hénin, T. Lelièvre, M. R. Shirts, O. Valsson and L. Delemotte, *Enhanced sampling methods for molecular dynamics simulations*, 2022.
71 J. Zhang, Y. Shi and P. Ren, *Protein-Ligand Interact.*, 2012, 99–120.
72 N. A. Mohamed, R. T. Bradshaw and J. W. Essex, *J. Comput. Chem.*, 2016, **37**, 2749–2758.
73 J. C. Wu, G. Chattree and P. Ren, *Theor. Chem. Acc.*, 2012, **131**, 1138.
74 M. M. Ghahremanpour, J. Tirado-Rives, M. Deshmukh, J. A. Ippolito, C.-H. Zhang, I. Cabeza de Vaca, M.-E. Liosi, K. S. Anderson and W. L. Jorgensen, *ACS Med. Chem. Lett.*, 2020, **11**, 2526–2533.
75 M. L. Laury, Z. Wang, A. S. Gordon and J. W. Ponder, *J. Comput.-Aided Mol. Des.*, 2018, **32**, 1087–1095.
76 Y. Shi, M. L. Laury, Z. Wang and J. W. Ponder, *J. Comput.-Aided Mol. Des.*, 2021, **35**, 79–93.
77 M. Amezcua, L. El Khoury and D. L. Mobley, *J. Comput.-Aided Mol. Des.*, 2021, **35**, 1–35.
78 D. R. Bell, R. Qi, Z. Jing, J. Y. Xiang, C. Mejias, M. J. Schnieders, J. W. Ponder and P. Ren, *Phys. Chem. Chem. Phys.*, 2016, **18**, 30261–30269.
79 M. Amezcua, J. Setiadi, Y. Ge and D. L. Mobley, *J. Comput.-Aided Mol. Des.*, 2022, **36**, 707–734.
80 M. Harger, D. Li, Z. Wang, K. Dalby, L. Lagardère, J.-P. Piquemal, J. Ponder and P. Ren, *J. Comput. Chem.*, 2017, **38**, 2047–2055.
81 H. S. Muddana, A. T. Fenley, D. L. Mobley and M. K. Gilson, *J. Comput.-Aided Mol. Des.*, 2014, **28**, 305–317.
82 T. Jaffrelot Inizan, F. Célerse, O. Adjoua, D. El Ahdab, L.-H. Jolly, C. Liu, P. Ren, M. Montes, N. Lagarde, L. Lagardère, P. Monmarché and J.-P. Piquemal, *Chem. Sci.*, 2021, **12**, 4889–4907.
83 P. Monmarché, J. Weisman, L. Lagardère and J.-P. Piquemal, *J. Chem. Phys.*, 2020, **153**, 024101.
84 L. Zhang, H. Wang, M. C. Muniz, A. Z. Panagiotopoulos, R. Car and W. E., *J. Chem. Phys.*, 2022, **156**, 124107.
85 N. T. P. Tu, N. Rezajooei, E. R. Johnson and C. Rowley, *Digital Discovery*, 2023, DOI: **10.1039/D2DD00150K**.
86 T. Plé, L. Lagardère and J.-P. Piquemal, *Force-Field-Enhanced Neural Network Interactions: from Local Equivariant Embedding to Atom-in-Molecule properties and long-range effects*, 2023, **https://arxiv.org/abs/2301.08734**.
87 P. P. Poier, L. Lagardère and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2022, **18**, 1633–1645.
88 P. P. Poier, T. Jaffrelot Inizan, O. Adjoua, L. Lagardère and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2022, **13**, 4381–4388.

Conclusion

This work introduces Deep-HP an highly parallel multi-GPU neural network platform integrated into Tinker-HP. Deep-HP empowers users to combine their MLP models with FFs through Tinker-HP, enabling simulations of millions of atoms and routine production simulations of millions of atom biomolecular systems with advanced neural network models like ANI-2X and recently developed equivariant neural network models. We have demonstrated the platform's capabilities by simulating large biologically-relevant systems, such as the SARS-CoV-2 M^{pro} and spike protein, on up to 124 GPUs using ANI-2X.

The platform's coupling of state-of-the-art PFFs, such as AMOEBA, with MLPs has led to the development of a hybrid MLP/PFFs potential. To enhance the hybrid model's capabilities, we designed multi-timestep integrators that significantly accelerate simulations compared to Velocity Verlet, achieving a speedup of 40 to 2 relative to AMOEBA in the ANI-2X/AMOEBA hybrid approach. This hybrid approach incorporates physically-motivated long-range effects and enables efficient simulations with Particle Mesh Ewald periodic boundary conditions, including polarizable counter ions. Additionally, it leverages ANI-2X's accurate description of ligand potential energy surfaces, allowing high-resolution exploration of conformational space through hybrid model MD simulations. These advancements enable the treatment of any type of ligand, including charged ones, and pave the way for routine long timescale simulations using NNPs/PFFs in million-atom biological systems, offering significant speedup compared to traditional ligand binding QM/MM simulations.

Here, we first assessed the hybrid model's accuracy by evaluating solvation free energies of 70 molecules with various functional groups, including charged ones, in three non-aqueous solvents and water. The hybrid model demonstrated comparable or superior accuracy to the AMOEBA polarizable force field. This achievement sets the stage for simulating complex biological processes with neural networks where environmental polarizability plays a crucial role.

Furthermore, we evaluated the performance of our hybrid model on binding free energies of 14 challenging host-guest systems from the SAMPL host-guest binding competitions. Despite the complexity of the chemical environments and the presence of charged ligands, our hybrid model outperformed AMOEBA. Overall, the ANI-2X/AMOEBA hybrid approach achieved an accuracy within the range of chemical accuracy, 0.94 kcal/mol, for solvation and more importantly for absolute binding free energies on the SAMPL challenge. Although further work is needed to assess the statistical uncertainties associated with such hybrid simulations, advancements in software and high-performance computing will facilitate this assessment in the coming years. Notably, while AMOEBA alone can achieve sub-kcal/mol accuracy in some cases, the hybrid neural network technology represents a step forward, as demonstrated on the SAMPL 6, 7, and 8 sets.

A total of 10μs of MD simulations with the hybrid model were performed in just a few days. Which represents the most extensive MD study incorporating neural networks conducted to date. These performance levels are expected to improve further through ongoing Deep-HP optimizations, neural network model research and advancements in GPU hardware.

Finally, Deep-HP serves as a platform that not only facilitates the implementation of the next generation of enhanced MLPs but also fosters their continued development. It offers direct coupling

of neural networks with physics-driven contributions that surpass the limitations of multipolar electrostatics and polarization by incorporating many-body dispersion models, such as the DNN-MBD model explained in the last section.[14, 15] With its primary objective of enabling large-scale production simulations using trained ML/hybrid models, Deep-HP is poised to expand its simulation capabilities to other system classes, including materials and catalysis applications. The availability of the Deep-HP computational platform paves the way for extensive hybrid MLP simulations, combining the strengths of neural networks and force fields, in the fields of biophysics and drug discovery.

2.3 Combining Nuclear Quantum Effects with Force Fields and Machine Learning Potentials with Quantum-HP

Introduction

In the preceding chapter, it was discussed that most models in computational chemistry assume that the nuclei are classical particles, disregarding NQEs or incorporating them implicitly through a parametrization procedure, leading to limited transferability. However, with the increasing accuracy and efficiency of MD and MLP simulations, the explicit inclusion of NQEs has become crucial. These effects are particularly significant in simulations of systems under extreme conditions, e.g low temperatures and high pressures, but also in standard conditions. Studying the effect of NQEs of biological processes is also of particular interest, as proton transfer is at the core of some phenomena. However, incorporating NQEs into these simulations also requires efficient and parallel implementations capable of simulating large systems and long timescales.

Highly efficient implementations of NQEs methods like PIMD or adQTB, as discussed in the previous chapter, are scarce in standard MD codes. Although a few parallel and open-source implementations of PIMD exist, such as i-PI [163] and LAMMPS [164], they have limitations. For instance, i-PI, while flexible and high-level is unsuitable for large MD simulations of biological systems. On the other hand, LAMMPS offers an efficient built-in implementation of PIMD but lacks flexibility in its parallel implementation, and is not compatible with PFFs like AMOEBA as it is mainly used for condensed matter physics. Therefore, a highly efficient PIMD and adQTB implementation compatible with state-of-the-art PFFs and MLPs for biological simulations is still lacking.

This work presents the implementation of Quantum-HP, a highly parallel platform for the explicit inclusion of NQEs within Tinker-HP, offering compatibility with multi-GPU acceleration. Quantum-HP is designed to integrate seamlessly with the recently introduced Deep-HP platform, as discussed in the previous section, which enables MD simulations with MLPs or hybrid MLP/PFF models.

Although the inclusion of NQEs within Tinker-HP was a marginal focus of this thesis, it remains a crucial aspect that deserves attention. The development of the Q-NN-AMOEBA model, which is extensively explained in the upcoming chapter, heavily relies on this platform.[76]

JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

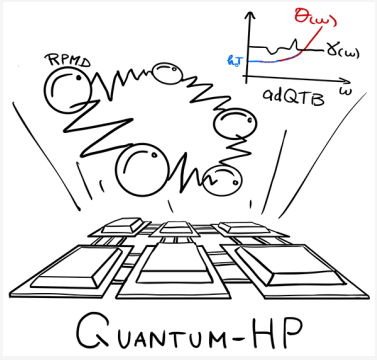
Routine Molecular Dynamics Simulations Including Nuclear Quantum Effects: From Force Fields to Machine Learning Potentials

Thomas Plé,* Nastasia Mauger, Olivier Adjoua, Théo Jaffrelot Inizan, Louis Lagardère,* Simon Huppert, and Jean-Philip Piquemal*

Cite This: https://doi.org/10.1021/acs.jctc.2c01233 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: We report the implementation of a multi-CPU and multi-GPU massively parallel platform dedicated to the explicit inclusion of nuclear quantum effects (NQEs) in the Tinker-HP molecular dynamics (MD) package. The platform, denoted Quantum-HP, exploits two simulation strategies: the Ring-Polymer Molecular Dynamics (RPMD) that provides exact structural properties at the cost of a MD simulation in an extended space of multiple replicas and the adaptive Quantum Thermal Bath (adQTB) that imposes the quantum distribution of energy on a classical system via a generalized Langevin thermostat and provides computationally affordable and accurate (though approximate) NQEs. We discuss some implementation details, efficient numerical schemes, and parallelization strategies and quickly review the GPU acceleration of our code. Our implementation allows an efficient inclusion of NQEs in MD simulations for very large systems, as demonstrated by scaling tests on water boxes with more than 200,000 atoms (simulated using the AMOEBA polarizable force field). We test the compatibility of the approach with Tinker-HP's recently introduced Deep-HP machine learning potentials module by computing water properties using the DeePMD potential with adQTB thermostatting. Finally, we show that the platform is also compatible with the alchemical free energy estimation capabilities of Tinker-HP and fast enough to perform simulations. Therefore, we study how NQEs affect the hydration free energy of small molecules solvated with the recently developed Q-AMOEBA water force field. Overall, the Quantum-HP platform allows users to perform routine quantum MD simulations of large condensed-phase systems and will help to shed new light on the quantum nature of important interactions in biological matter.



1. INTRODUCTION

Molecular dynamics (MD) is a powerful simulation tool that allows one to compute properties of atomistic systems in a wide range of conditions, with the aim of explaining experimental results or even be predictive. Over the last decades, it has been a very active field of research. Long and accurate simulations of large condensed-phase systems are now reachable with recent advances in high performance computing (HPC) and GPU acceleration. We can distinguish efforts made in this field in two categories: (a) improvements of the models for interatomic interactions and (b) more efficient and accurate simulations of the nuclear motion in the desired statistical ensemble. Regarding the first category, considerable improvements have been made in two directions: efficiency of first principle descriptions (for example, using Born−Oppenheimer density functional theory) on the one hand and accuracy of effective models (classical force fields,[1−3] polarizable force fields,[4−7] machine learning (ML) force fields[8−10]) on the other hand.

Regarding the second category, lots of attention has been given to the development of efficient integration schemes (multi-timestepping,[11,12] hybrid Monte Carlo algorithms[13,14]) or improved sampling methods (parallel tempering,[15] meta-dynamics[16]) in order to tackle the need for long simulations in complex energy landscapes. Most implementations, however, assume that the nuclei are classical particles, thus completely neglecting nuclear quantum effects (NQEs) or implicitly including them in an uncontrolled manner—for example, by fitting force fields on experimental data simulated using classical MD—which limits transferability.[17−19]

As MD simulations grow in accuracy and efficiency, the need for the explicit inclusion of NQEs becomes more and more apparent, be it in simulations of systems in extreme conditions (low temperatures, high pressures) where they can be massive[20−24] or even in more standard conditions where it has already been shown that more subtle NQEs are at play.[25−28] NQEs can be explicitly included in MD simulations in the framework of path integrals (PIMD) which provides an exact description of structural NQEs.[29,30] Even though they are

Received: December 6, 2022

considered as the gold standard, PIMD calculations are usually expensive as they require one to simulate the system in an extended phase space which size grows when NQEs are more pronounced. Cheaper approximate methods have been recently developed,[31−34] among which is the adaptive quantum thermal bath (adQTB)[35,36] that proved to be an accurate alternative to PIMD at the cost of a classical MD simulation.[37] As NQEs are suspected to play a role in some biological processes,[38,39] an efficient and parallel implementation of these methods is required to simulate the large systems and long time scales involved in such processes. As highlighted in several previous papers,[17,40,41] it is also desirable to design advanced force fields (FFs) with explicit NQEs from scratch (i.e., that do not implicitly incorporate them through parametrization). This endeavor, which is already challenging in a classical framework, was up to now nearly unachievable as it requires numerous quantum simulations to adapt the parameters of the model and because highly efficient implementations of PIMD or adQTB in standard MD codes are scarce. While the adQTB is, to our knowledge, not yet available in any massively parallel MD code, a few parallel and open-source implementations of PIMD are available. Among them, we mention in particular i-PI[42] and LAMMPS[43] (fix_pimd). i-PI is a high-level and flexible Python implementation that provides most "flavors" of PIMD. It however assumes that the evaluation of atomic forces (that is externalized via a socket system) largely dominates the computation time. This assumption is valid when using *ab initio* forces but not when using effective models—especially when their implementation is highly optimized and GPU accelerated—so that i-PI is not well suited for large MD simulations of biological systems. On the other hand, LAMMPS provides an efficient built-in implementation of PIMD but does not provide accelerated PI methods (such as ring-polymer contractions for example[44,45]) and, more importantly, is not yet compatible with advanced polarizable force fields such as AMOEBA.[46] A highly efficient PIMD and adQTB implementation compatible with state-of-the-art polarizable force fields for biological simulations is thus still lacking and highly desirable.

In this work, we report the implementation of Quantum-HP, a highly parallel platform for the explicit inclusion of NQEs, compatible with multi-GPU acceleration, inside the Tinker-HP molecular dynamics package.[47,48] The platform is fully compatible with all the force fields present in Tinker-HP, including classical ones (CHARMM, AMBER) and AMOEBA,[46,49,50] AMOEBA+,[51,52] and SIBFA[53,54] polarizable FFs, and allows the simulation of million-atom systems in a distributed architecture. The paper is organized as follows: Section 2 briefly describes the theory for the two methods that we implemented, namely, ring-polymer MD and the adQTB. Section 3 provides some important implementation details for both methods, including time integrators and parallelization strategies. Scaling and efficiency tests are also provided, as well as a brief description of the GPU acceleration. We briefly show in Section 4 that Quantum-HP is compatible with the new Deep-HP[55] platform that allows one to perform molecular dynamics using machine learning (ML) potentials or hybrid ML/MM force fields. Finally, in Section 5, we demonstrate the capabilities of the platform and the accuracy of the recently developed Q-AMOEBA force field[41] by computing the hydration free energies of a benchmark data set of small organic molecules, for which we obtain state-of-the-art accuracy when including NQEs using the adQTB. Section 6 provides some concluding remarks and outlooks for future developments and applications.

2. METHODS

In this section, we briefly describe the theoretical framework of the two methods for the inclusion of NQEs, namely, Ring-Polymer Molecular Dynamics (RPMD) and the adaptive Quantum Thermal Bath (adQTB), that we implemented in Tinker-HP.

2.1. Ring-Polymer Molecular Dynamics. Ring-Polymer Molecular Dynamics[56,57] is based on the imaginary-time path integral formulation of quantum statistical mechanics. This formalism allows one to express the canonical partition function $Z = \mathrm{Tr}[e^{-\beta\hat{H}}]$ of a quantum system (made of distinguishable particles at thermal equilibrium) as the one of an effective classical system. This system takes the form of a so-called "ring polymer" (as schematically depicted in Figure 1) where "beads" along the polymer are replicas of the whole

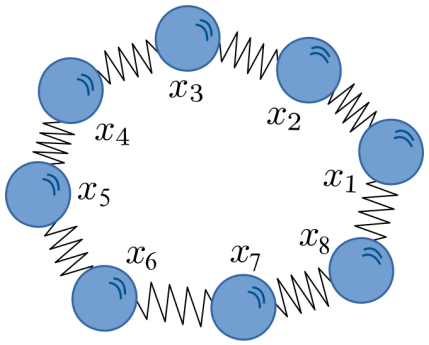


Figure 1. Schematic representation of the ring-polymer path integral for $\nu = 8$. Each bead $x_1, ..., x_\nu$ (represented by a blue circle) is subject to the physical potential and connected to its nearest neighbors via a harmonic potential (represented as springs).

original system (independently subject to the interatomic potential V) that interact through a harmonic potential. In particular, in our implementation, we employ the scaled normal mode representation of the ring polymer that describes it in terms of a center of mass (called the centroid) and fluctuations around it. In this framework, the quantum partition function is written as

$$Z = \lim_{\nu\to\infty} \int \mathrm{d}Q e^{-\beta(U_\nu(Q)+\sum_{n>0}\frac{1}{2}\omega_n^2 Q_n^T M Q_n)} \tag{1}$$

where $Q = (Q_0, ..., Q_{\nu-1})$ are the amplitudes of the ν modes describing the ring polymer (each being a vector of size $3N_{\mathrm{atoms}}$), M is the diagonal mass matrix of the physical system, and ω_n are the characteristic frequencies of the normal modes which are defined as the square roots of the eigenvalues (ordered by increasing amplitude) of the $\nu \times \nu$ matrix:

$$\Omega_\nu^2 = \frac{\nu^2}{\hbar^2\beta^2}\begin{pmatrix} 2 & -1 & & & -1 \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ -1 & & & -1 & 2 \end{pmatrix} \tag{2}$$

where all the undefined terms in the matrix are zeros. The normal modes are subject to the potential $U_\nu(Q)$ which is defined as

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

$$U_\nu(Q) = \frac{1}{\nu}\sum_{i=0}^{\nu-1} V(x_i^{(\nu)}(Q)) \tag{3}$$

where V is the physical interatomic potential, and $x_i^{(\nu)}$ is the position of the ith bead of the ring polymer, that is constructed from the ν normal mode amplitudes as

$$x_i^{(\nu)}(Q) = Q_0 + \sqrt{\nu}\sum_{n=1}^{\nu-1} T_{in}^{(\nu)} Q_n \tag{4}$$

with $T^{(\nu)}$ the unitary transfer matrix which columns are the eigenvectors of the matrix (eq 2). We note that Q_0 represents the position of the centroid of the ring polymer (with associated frequency $\omega_0 = 0$) and that $Q_{n>0}$ are called fluctuation modes. From eq 1, we write the probability distribution of the ring polymer $\rho_\nu(Q)$ as

$$\rho_\nu(Q) = \frac{1}{Z_\nu} e^{-\beta(U_\nu(Q) + \sum_{n>0}\frac{1}{2}\omega_n^2 Q_n^T M Q_n)} \tag{5}$$

with Z_ν a normalization constant such that $Z = \lim_{\nu\to\infty} Z_\nu$. In this framework, the thermal equilibrium average of any position-dependent observable $A(\hat{x})$ is obtained as an average over the distribution ρ_ν (in the limit $\nu \to \infty$):

$$\langle A(\hat{x})\rangle_\beta = \mathrm{Tr}\left[A(\hat{x})\frac{e^{-\beta\hat{H}}}{Z}\right] = \lim_{\nu\to\infty}\int \mathrm{d}Q A_\nu(Q)\rho_\nu(Q) \tag{6}$$

with $A_\nu(Q) = \sum_{i=0}^{\nu-1} A(x_i^{(\nu)}(Q))/\nu$ defined similarly as in eq 3 for the potential energy.

In order to perform molecular dynamics simulations, a set of momenta $\mathcal{P} = (P_0, ..., P_{\nu-1})$ are associated with the normal modes so that the joint probability density becomes

$$\rho_\nu(Q, \mathcal{P}) \propto \rho_\nu(Q) e^{-\beta\sum_n \frac{1}{2}P_n^T M^{-1} P_n} \tag{7}$$

This formalism also allows one to compute approximate (Kubo-transformed) time correlation functions of position-dependent observables as[58,59]

$$K_{AB}^{(\nu)}(t) = \int \mathrm{d}Q\mathrm{d}\mathcal{P}\rho_\nu(Q, \mathcal{P})A_\nu(Q)B_\nu(Q(t)) \tag{8}$$

where $Q(t)$ is obtained by propagating for a duration t the ring-polymer equations of motion:

$$\begin{cases} \dot{Q}_n &= M^{-1}P_n \\ \dot{P}_n &= f_n(Q) - \omega_n^2 M Q_n \end{cases} \qquad (n = 0, ..., \nu - 1) \tag{9}$$

with $f_n(Q)$ the interatomic force projected on the nth normal mode which is obtained from the chain rule as

$$f_0(Q) = -\frac{1}{\nu}\sum_{i=0}^{\nu-1}\nabla V(x_i^{(\nu)}(Q)) \tag{10}$$

$$f_{n>0}(Q) = -\frac{1}{\sqrt{\nu}}\sum_{i=0}^{\nu-1} T_{in}^{(\nu)}\nabla V(x_i^{(\nu)}(Q)) \tag{11}$$

Importantly, we note that while eq 6 is exact in the $\nu \to \infty$ limit independently of the form of V, this is not the case for eq 8 as the dynamics of the ring polymer generated by the equations of motion (eq 9) does not generally reproduce the exact quantum dynamics.[60,61] This approximation was however shown to be quite robust and to provide relevant results in many applications.[57,62,63]

2.2. Adaptive Quantum Thermal Bath. The adaptive Quantum Thermal Bath (adQTB) is an hybrid quantum-classical method that relies on a generalized Langevin thermostat in order to impose the quantum distribution of energy on a classical system:[32,35]

$$\begin{cases} \dot{x} &= M^{-1}p \\ \dot{p} &= -\nabla V(x) - \gamma_0 p + F(t) \end{cases} \tag{12}$$

where γ_0 is a friction coefficient, and $F(t)$ is a colored random force with the following power spectrum:

$$C_{F_iF_j}(\omega) = 2m_i\gamma_i(\omega)\Theta(\omega, \beta)\delta_i^j \quad (i, j = 1, ..., 3N_{\mathrm{atoms}}) \tag{13}$$

with m_i the ith diagonal element of the mass matrix M, δ_i^j the Kroneker delta symbol, and

$$\Theta(\omega, \beta) = \frac{\hbar\omega/2}{\tanh(\beta\hbar\omega/2)} \tag{14}$$

the average thermal energy of a quantum harmonic oscillator of frequency ω at inverse temperature β. The parameters $\gamma_i(\omega)$ in the random force amplitude are adjusted in order to minimize the average deviation from the quantum fluctuation–dissipation theorem[35,36,64] (FDT):

$$\begin{aligned}\gamma_i^\star(\omega) &= \arg\min_{\gamma_i(\omega)} |\Delta_{\mathrm{FDT},i}(\omega)| \\ &= \arg\min_{\gamma_i(\omega)} |m_i\gamma_i(\omega)C_{v_iv_i}(\omega) - \mathrm{Re}[C_{v_iF_i}(\omega)]|\end{aligned} \tag{15}$$

where $C_{v_iv_i}(\omega)$ (respectively, $C_{v_iF_i}(\omega)$) is the velocity–velocity (respectively, velocity–random force) correlation spectrum estimated in a QTB simulation using the trial parameter γ_i in the random force power spectrum. The optimum $\Delta_{\mathrm{FDT},i}(\omega) = 0$ indicates that the thermal energy (including zero-point energy) is correctly distributed in the system, according to the quantum FDT.

For a purely harmonic system, $\gamma_i^\star(\omega)$ is known analytically, and one can show that $\Delta_{\mathrm{FDT},i}(\omega) = 0$ for constant $\gamma_i^\star(\omega) = \gamma_0, \forall\omega$. Additionally, in this particular case, the QTB dynamics produces the exact quantum phase space distribution for sufficiently small values of γ_0.[65] The original QTB, as devised in ref 32 is obtained by using the harmonic solution $\gamma_i(\omega) = \gamma_0$, even for anharmonic systems. Deviations from the quantum FDT might therefore be present, that manifest through the well-documented ZPE leakage.[66,67]

The fluctuation–dissipation theorem provides a generic criterion to optimize the parameters for any anharmonic system, and no *a priori* information on the system is required. Deviations $\Delta_{\mathrm{FDT},i}(\omega)$ from the quantum FDT are estimated along the dynamics and used to adapt the adQTB parameters $\gamma_i(\omega)$ with a procedure detailed in Section 3.2.2. In practice, the estimator of $\Delta_{\mathrm{FDT},i}(\omega)$ is subject to statistical noise so that we do not strictly optimize the parameters but rather let them fluctuate around their optimal value such that $\Delta_{\mathrm{FDT},i}(\omega)$ should fluctuate around zero. The adaptation procedure is performed in an equilibration phase which duration typically ranges from a few picoseconds to a few hundred picoseconds depending on the system.

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

While the adQTB cannot be formally derived from first principles, it was recently shown to provide accurate results even in very anharmonic systems.[37] As its computational cost is essentially the same as that of a classical MD simulation, it is a promising approach for the quantum simulation of large biological systems. It was the method of choice for the recent development of the Q-AMOEBA force field,[41] and we show in Section 5 that the combination Q-AMOEBA/adQTB can be used to accurately compute hydration free energies of small organic molecules.

3. IMPLEMENTATION

This section provides implementation details for both methods, starting with RPMD and following with adQTB. Integration schemes and parallelization strategies are discussed, as well as some technical points specific to each method. We describe here the implementation for simulations in the NVT ensemble. For both adQTB and PIMD, constant-pressure simulations are implemented via the Langevin piston method[68] following ref 69. To conclude this section, some scaling tests (using the AMOEBA polarizable FF) are presented in order to compare the efficiency of the quantum methods compared to reference classical MD calculations.

3.1. RPMD. *3.1.1. Integration Scheme.* In order to sample the canonical distribution of the ring polymer (7), we attach a Langevin thermostat to each normal mode. The equations of motion are then integrated using the BAOAB scheme, originally introduced by Leimkhuller et al.[70] and adapted for path-integral simulations (following, for example, ref 71). The choice of the normal mode representation allows one to efficiently integrate the rapidly oscillating motion due to the path-integral harmonic chain and to use a simulation time step that is essentially dictated by the characteristic time scales of the interatomic potential V (and is thus similar to classical simulations). Our implementation also utilizes the TRPMD scheme of ref 72 in which we apply a strong (critically damped) Langevin thermostat to the fluctuation modes. In order to ensure ergodicity while minimizing the disruption to the dynamics, we apply an underdamped Langevin thermostat to the centroid (the original TRPMD is thus recovered in the limit of zero damping on the centroid). The TRPMD equations of motion read as follows:

$$\begin{pmatrix}\dot{Q}_n\\ \dot{P}_n\end{pmatrix} = \overbrace{\begin{pmatrix}M^{-1}P_n\\ -\omega_n^2 M Q_n\end{pmatrix}}^{e^{i\mathcal{L}_A t}} + \overbrace{\begin{pmatrix}0\\ f_n(Q)\end{pmatrix}}^{e^{i\mathcal{L}_B t}} + \underbrace{\begin{pmatrix}0\\ -\gamma_n P_n + (2\gamma_n\beta^{-1}M)^{1/2}R_n(t)\end{pmatrix}}_{e^{i\mathcal{L}_O t}} \tag{16}$$

with $R_n(t)$ a $3N_{\text{atoms}}$ vector of uncorrelated standard Gaussian white noise, and $\gamma_n = \max(\gamma_0, \omega_n)$ the (critical) friction coefficient for each normal mode. We decompose the equations of motion into three analytically solvable blocks which formal solutions are denoted by the corresponding Liouville propagators $e^{i\mathcal{L}_A t}$, $e^{i\mathcal{L}_B t}$, and $e^{i\mathcal{L}_O t}$. The propagator $e^{i\mathcal{L}_A t}$ corresponds to a harmonic evolution of the fluctuation modes and a simple translation of the centroid position (since $\omega_0 = 0$). The propagator $e^{i\mathcal{L}_B t}$ is a translation of the momenta according to the interatomic forces (projected on the normal modes) and the propagator $e^{i\mathcal{L}_O t}$ is a standard Ornstein–Uhlenbeck process[73] for each normal mode. The full time propagator over a duration $t = n_{\text{step}}\Delta t$ is then symmetrically broken up as

$$e^{i\mathcal{L}_{\text{RPMD}}t} \approx \left(e^{i\mathcal{L}_B\frac{\Delta t}{2}}e^{i\mathcal{L}_A\frac{\Delta t}{2}}e^{i\mathcal{L}_O\Delta t}e^{i\mathcal{L}_A\frac{\Delta t}{2}}e^{i\mathcal{L}_B\frac{\Delta t}{2}}\right)^{n_{\text{step}}} \tag{17}$$

where Δt is the simulation time step. The implementation also optionally allows the use of the BCOCB variant recently introduced in ref 74 where the exact integration of $e^{i\mathcal{L}_A\frac{\Delta t}{2}}$ is replaced by a numerical scheme that allows for a better stability of the dynamics and larger timesteps (a 3-fold increase in some cases) when a large number of beads is required.

3.1.2. Multi-Timestep Methods. For interatomic potentials of the form

$$V = V_s + V_f \tag{18}$$

with V_s a slowly varying and expensive component of the total potential (nonbonded interactions in the case of AMOEBA) and V_f a quickly varying and inexpensive component (bonded interactions in the case of AMOEBA), simulations can be made more efficient using multiple timestepping to compute the expensive V_s less frequently. In the case of RPMD, the splitting (eq 18) can be used to our advantage both in the time integration scheme (RESPA algorithm[11]) and in the computation of the interatomic forces on the ring-polymer beads (ring-polymer contraction).

Path-Integral RESPA Integrator. To improve the efficiency of the integration scheme, we use the BAOAB-RESPA multitimestep algorithm, which was initially designed for classical Langevin MD[11,12] and adapted to path-integrals simulations.[75−77] For this method, the B block (update of the velocities according to the interatomic forces) is split into Bf (associated with ∇V_f) and Bs block (associated with ∇V_s) and the dynamics is propagated using a two-stage symmetric Trotter breakup of the full time-propagator:

$$e^{i\mathcal{L}_{\text{RPMD}}t} \approx \left(e^{i\mathcal{L}_{Bs}\frac{n_{\text{alt}}\Delta t}{2}}\left(e^{i\mathcal{L}_{Bf}\frac{\Delta t}{2}}e^{i\mathcal{L}_A\frac{\Delta t}{2}}e^{i\mathcal{L}_O\Delta t}e^{i\mathcal{L}_A\frac{\Delta t}{2}}e^{i\mathcal{L}_{Bf}\frac{\Delta t}{2}}\right)^{n_{\text{alt}}}e^{i\mathcal{L}_{Bs}\frac{n_{\text{alt}}\Delta t}{2}}\right)^{n_{\text{step}}} \tag{19}$$

This expression shows that, for a full time step of integration, an inner loop of n_{alt} BfAOABf timesteps is performed with a short time step Δt. For every n_{alt} time step, a propagation of the block Bs is performed with the larger time step $n_{\text{alt}}\Delta t$. When the computation of ∇V_s dominates the total calculation time, this scheme allows performance gains of up to a factor n_{alt}. Typically, the smaller time step Δt ranges between 0.2 fs and 1 fs while the larger time step $n_{\text{alt}}\Delta t$ is of the order of 2 fs.

Ring-Polymer Contractions. Taking further advantage of the separation of the interatomic potential $V = V_s + V_f$ used in the RESPA integrator, we implemented the ring-polymer contraction (RPC) scheme introduced in refs 44 and 45. This scheme is based on the assumption that the motion of high-frequency normal modes of the ring polymer is only weakly affected by the slowly varying interatomic forces, so that one can neglect these modes when evaluating the slow forces. This allows one to evaluate the slowly varying potential on a "contracted" set of beads instead of the full ring polymer:

$$U_\nu(Q) \approx \frac{1}{\nu}\sum_{i=0}^{\nu-1} V_f(x_i^{(\nu)}(Q)) + \frac{1}{\tilde{\nu}}\sum_{i=0}^{\tilde{\nu}-1} V_s(x_i^{(\tilde{\nu})}(Q)) \tag{20}$$

with $\tilde{\nu} \le \nu$ and where the coordinates $x_i^{(\tilde{\nu})}$ are computed similarly as in eq 4 but considering only the $\tilde{\nu}$ lowest-frequency

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

normal modes. When $\tilde{\nu} = \nu$, the full ring-polymer potential is recovered. On the other hand, when $\tilde{\nu} = 1$, V_s is only evaluated at the centroid of the ring polymer. In practice, $\tilde{\nu}$ is an additional convergence parameter that must be checked for each system. As demonstrated in refs 78 and 79, the RPC scheme can lead to large gains in performance for some systems. For example, in the case of liquid water modeled via the AMOEBA potential, accurate simulations can be achieved with $\nu = 32$ for the bonded interactions, and only $\tilde{\nu} \approx 5$ for the nonbonded interactions. As the nonbonded interactions are much more expensive to compute, this leads to a significant gain in performance. The RPC scheme is of course compatible with the RESPA integrator which further reduces the number of required evaluations of the slowly varying forces.

3.1.3. Massively Parallel Implementation. The most time-consuming operation in a MD simulation is usually the evaluation of the interatomic forces. It is even more marked in path-integral simulations, where the forces must be evaluated on multiple replicas of the system. However, this evaluation is independent for each bead $x_i^{(\nu)}(Q)$, and it is thus efficient to parallelize by assigning the evaluation of the forces on each bead to a different process (or set of processes). When the total number of processes N_{proc} is smaller than the number of beads, each process independently evaluates the forces on a subset of the replicas, as depicted in the top part of Figure 2.

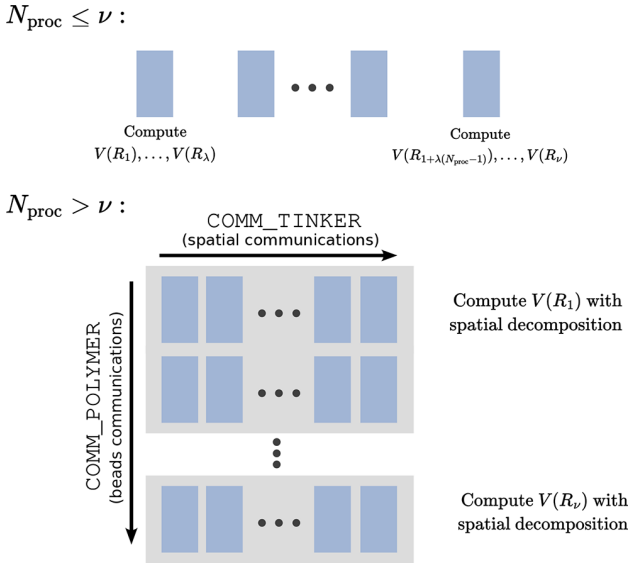


Figure 2. Schematic representation of the parallelization scheme used for the evaluation of the forces in RPMD simulations. The figure distinguishes the two subcases: $N_{\text{proc}} \leq \nu$ (top) and $N_{\text{proc}} > \nu$ (bottom). In the top figure, we define $\lambda = \nu/N_{\text{proc}}$.

On the other hand, when $N_{\text{proc}} > \nu$, we employ a two-level parallelization scheme that leverages the spatial domain decomposition already implemented in Tinker-HP.[47] To this aim, the main MPI communicator is split into a grid as schematically shown in the bottom part of Figure 2. The communicator COMM_POLYMER (of size $N_{\text{proc}}^{\text{polymer}}$) allows communication between different beads within the same spatial region, while the communicator COMM_TINKER (of size $N_{\text{proc}}^{\text{spatial}}$), that runs horizontally in the figure, allows for communication between different spatial regions at a fixed bead index.

Once interatomic forces have been evaluated on each bead, they are communicated through COMM_POLYMER and projected on the normal modes according to eq 11. At this point, the equations of motion for each atom (and each ring-polymer normal mode) can be independently propagated until the next force evaluation is required. This propagation is parallelized by evenly distributing the local atoms among the $N_{\text{proc}}^{\text{polymer}}$ processes of each spatial region. When the centroid of the ring polymer of an atom changes domains, the information for all its normal modes are transferred to the neighboring processing units. Neighbor lists are also computed with respect to centroid positions: if the centroids of two atoms are considered neighbors, all the corresponding beads are also considered neighbors. This avoids duplicating neighbor lists for all the beads, thus drastically reducing the associated computational cost and memory requirements.

Note that when using the RPC scheme of Section 3.1.2, the parallelization strategy is defined based on the number $\tilde{\nu}$ of beads in the contracted ring polymer instead of the full number ν. The evaluation of the slowly varying forces (typically the most time-consuming step of the calculation) is then distributed for the contracted ring polymer with the same parallelization strategy as in Figure 2, while for the evaluation of the quickly varying forces, the beads of the full ring polymer are partitioned using the same spatial decomposition as for the contracted one.

3.2. adQTB. The adQTB implementation uses the standard classical Langevin integrators (BAOAB, BAOAB-RESPA, BAOAB-RESPA1) previously included in Tinker-HP and only replaces the white noise random forces by the adQTB colored noise. However, contrary to white noise that can easily be generated on the fly using a standard pseudorandom number generator, colored noise is not memoryless. To generate numerical noise with the adequate memory kernel, the trajectory is split into segments of N_{seg} timesteps (typically, $N_{\text{seg}} \sim 1000$). At the end of each segment, the adaptation procedure is performed, and the colored noise is generated in advance to be used in the next segment.

3.2.1. Colored Noise Generation. We generate the adQTB colored noise following the *segmented* procedure described in the appendix of ref 36. In a nutshell, a random force with autocorrelation given by eq 13 is computed by performing a convolution between a normalized white noise and the Fourier transform of the square root of eq 13 (with corrections for finite time step[35] and nonzero friction[37]). In practice, the convolution is performed in Fourier space (using a standard FFT library) at the beginning of each segment. Note that in the *segmented* procedure, one needs to store $3N_{\text{seg}}$ white noise random numbers for each degree of freedom in order to ensure that the colored noise memory is consistent between segments. Figure 3 shows a schematic flowchart of the adQTB integration scheme, in which the different steps of the *segmented* noise generation procedure are briefly outlined.

3.2.2. Computation of the adQTB Spectra and Adaptation Procedure. As explained in details in refs 35 and 36, the adaptive QTB relies on the quantum fluctuation−dissipation theorem to monitor and compensate ZPE leakage. To that end, we evaluate the deviations from the FDT defined for each degree of freedom i as

$$\Delta_{\text{FDT},i}(\omega) = m_i C_{v_i v_i}(\omega)\gamma_i(\omega) - \text{Re}[C_{v_i F_i}(\omega)] \tag{21}$$

The correlation functions are estimated at the end of each segment from the trajectories of v_i and F_i:

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

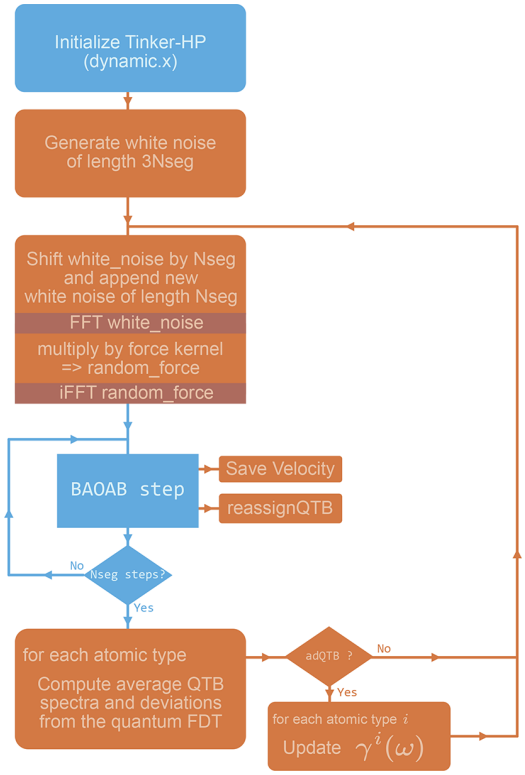


Figure 3. Flowchart of a molecular dynamics simulation using the adQTB thermostat.

$$C_{v_i v_i}(\omega) \propto |\tilde{v}_i(\omega)|^2$$
$$C_{v_i F_i}(\omega) \propto \tilde{v}_i(\omega)\tilde{F}_i^*(\omega) \tag{22}$$

where $\tilde{v}_i(\omega)$ and $\tilde{F}_i(\omega)$ are the (discrete) Fourier transforms of the trajectories $v_i(t)$ and $F_i(t)$ over the last segment (the last N_{seg} timesteps, that have thus to be stored in memory). In practice, the values of ω are discretized consistently with the discrete Fourier transform over the segment, though for simplicity, we keep the continuous notation for ω in the following.

In principle, the adjustable parameters of the bath $\gamma_i(\omega)$ could be optimized for each degree of freedom in order to cancel each $\Delta_{FDT,i}(\omega)$. In practice, we set the same $\gamma_i(\omega)$ for all degrees of freedom that share the same atom type number z in Tinker's input parameters, and optimize using the averaged $\overline{\Delta}_{FDT,z} = \frac{1}{N_z}\sum_{i\in z}\Delta_{FDT,i}$. This allows one to average statistical fluctuations that may affect $\Delta_{FDT,i}$ over all equivalent degrees of freedom, thus improving the convergence of the adaptation procedure. The implementation provides two adaptation schemes. In the first scheme, denoted as SIMPLE and described in details in ref 35, the coefficients are adapted at the end of each segment according to

$$\gamma_z^{(k+1)}(\omega) = \gamma_z^{(k)}(\omega) - A_{\gamma,z}\overline{\Delta}_{FDT,z}^{(k)}(\omega) \tag{23}$$

where $\overline{\Delta}_{FDT,z}^{(k)}$ is computed from the kth segment of trajectory, and the $\gamma_z^{(k)}$ are the corresponding bath parameters, while $\gamma_z^{(k+1)}$ are the new parameters to be used in the next segment. The coefficients $A_{\gamma,z}$ allow one to adjust the adaptation speed, for each atom type z. The second scheme, denoted as RATIO, allows for a faster adaptation of the bath parameters when large numbers of atoms share the same type z, while maintaining a controllable level of noise on γ_z. This adaptation scheme is based on the fact that, if $\gamma_z^\star(\omega)$ were the optimal parameters, we would have

$$\overline{\Delta}_{FDT,z}^\star(\omega) = 0 \Leftrightarrow \gamma_z^\star(\omega) = \frac{\mathrm{Re}[\overline{C}_{vF,z}(\omega)]}{m_z\overline{C}_{vv,z}(\omega)} \tag{24}$$

where $\overline{C}_{vF,z}$ and $\overline{C}_{vv,z}$ are defined similarly as $\overline{\Delta}_{FDT,z}$. Thus, for each type z, we define the new parameters as

$$\gamma_z^{(k+1)}(\omega) = \frac{\mathrm{Re}[\overline{C}_{vF,z}^{(k)}(\omega)]}{m_z\overline{C}_{vv,z}^{(k)}(\omega)} \tag{25}$$

The optimal value of $\gamma_z(\omega)$ should then be a fixed point of this iterative scheme. It should be noted that, due to numerical noise in the estimators of $\overline{C}_{vF,z}^{(k)}$ and $\overline{C}_{vv,z}^{(k)}$, the estimator of $\gamma_z^\star(\omega)$ resulting from the iterative process may be affected by large fluctuations and possibly biased. To fix this issue, we replace the ratio in eq 25 by a ratio of spectra obtained from a running average with an exponentially decaying window. For example, for $\overline{C}_{vv,z}$:

$$\langle\overline{C}_{vv,z}\rangle_{\tau_z}^{(k)} = \langle\overline{C}_{vv,z}\rangle_{\tau_z}^{(k-1)}e^{-N_{seg}\Delta t/\tau_z} + \overline{C}_{vv,z}^{(k)}(1 - e^{-N_{seg}\Delta t/\tau_z}) \tag{26}$$

The parameters τ_z then dictate the adaptation speed, and their admissible values critically depend on the level of statistical noise on both spectra, i.e., on the number of equivalent degrees of freedom on which they are averaged. As an example, when simulating a large box of liquid water, where all H and all O atoms are equivalent on average, values of τ_O and τ_H of the order of 100 fs to 1 ps are sufficient to provide an accurate and fast adaptation (yielding the same parameters as a slow adaptation with the SIMPLE method). On the other hand, when simulating an isolated molecule for which the spectra can only be averaged on few atoms, longer adaptation times are required with τ_z typically of the order of 100 ps. Note that it is possible to combine both adaptation methods, for example, by using the SIMPLE method to slowly adapt the parameters of a solute molecule while quickly adapting the parameters of the solvent with the RATIO scheme.

Finally, in order for the random force power spectrum to be well defined, a lower bound γ_{min} is set on γ_z by performing the operation $\gamma_z^{(k+1)}(\omega) \leftarrow \max(\gamma_{min}, \gamma_z^{(k+1)}(\omega))$ before generating a new segment of colored noise. By default, we set $\gamma_{min} = 0.01\gamma_0$. As illustrated in ref 35, this lower bound implies that the ZPE leakage cannot be compensated with an arbitrarily small value of the friction coefficient γ_0.

3.2.3. Massively Parallel Implementation. The parallelization scheme for the adQTB is straightforward as it fully utilizes the spatial decomposition previously implemented in Tinker-HP. The only additional burden compared to classical dynamics is the necessity to keep track of the colored noise for each degree of freedom. Indeed, when an atom is transferred to another cell of the spatial decomposition, its pregenerated colored noise must also be transferred. This corresponds to the "reassignQTB" of Figure 3. In order to avoid unnecessary communications, we ensure that colored noise transfer between two processes happens at most once per segment for each atom.

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

At the end of each segment, the spectra in eq 21 are computed in parallel and averaged for each atom type z on the process of rank zero. The latter then performs the adaptation of the $\gamma_z(\omega)$ as described in Section 3.2.2 and broadcasts the updated parameters to the other processes so that each can then generate the new segment of colored noise for the atoms in its spatial decomposition region.

3.3. Extension to GPU Architectures. Additionally to the massively parallel MPI CPU version, we implemented both methods in the multi-GPU version of Tinker-HP. The critical part of the GPU acceleration, described in ref 48 is contained in the calculation of the interatomic forces and did not require any alterations. The GPU port of our methods was done through OpenACC directives in order to offload the generation of the colored noise for adQTB and the integration and normal modes calculations for RPMD onto the device. Much care was taken to suppress unnecessary data transfer between CPU and GPU so that all extra variables (positions and momenta of the normal modes in the case of RPMD and storage of noise and trajectory segments for adQTB) are GPU-resident, i.e., are uploaded once on the GPU at the beginning of the simulation and accessed almost exclusively by the GPU. In the case of multi-GPU calculations, direct GPU-to-GPU MPI communications are performed whenever the host architecture allows it.

3.4. Scaling and Efficiency Tests. We tested the parallelization efficiency on boxes of water of sizes 96,000 atoms (puddle) and 288,000 atoms (pond) simulated using the AMOEBA polarizable force field. Calculations were performed on the Joliot−Curie cluster located at TGCC and managed by the CEA. We used two of its partitions made of interconnected nodes. Traditional nodes from the first partition are made of two AMD Epyc processors with 64 cores each and clocked at 2.6 Ghz. The second partition holds two CPUs Intel Cascade Lake of 20 cores each, clocked at 2.1 GHz, and accelerated with four GPUs NVIDIA V100 interconnected with NVIDIA NVLink. All simulations use a time step of 0.2 fs, which safely ensures a low integration error for all methods. Figure 4 shows performance (measured in nanoseconds of simulation per day of computation) as a function of the number of CPUs in a log−log scale for both system sizes. We first notice that the performances of the adQTB are almost identical to that of classical MD, confirming that the colored noise generation and the adaptation scheme only make a small contribution to the computation time for these moderately large systems. The raw performance of the RPMD is of course lower than that of classical MD (due to the 32 replicas used for the simulation), but the scaling with the number of processes is similar. Note that for these simulations there are more processes than RPMD replicas so that the two-level bead/spatial parallelization described in Section 3.1.3 is fully used.

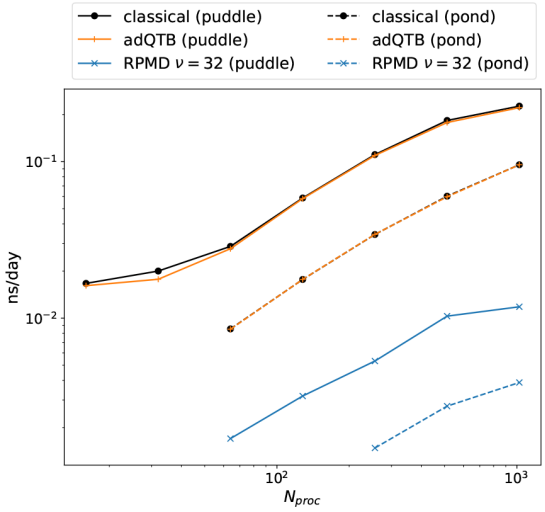


Figure 4. Scaling tests on multi-CPU architecture for the different methods with the AMOEBA polarizable force field. Performance is indicated by the number of nanoseconds of simulation per walltime day as a function of the number of processes.

Figure 5 shows the performance of the same methods on multi-GPU architecture. Again, we obtain very similar performances in classical MD and in adQTB and a very significant performance increase compared to the CPU architecture. The drop in performance when going from four to eight GPUs is due to inefficiencies in the out-of-node communications (nodes at TGCC are composed of four interconnected V100 GPUs) which have a critical impact for the spatial decomposition parallelization scheme. On the other hand, multinode parallelization in RPMD remains very efficient as long as the number of GPUs is smaller than the number of replicas since the interatomic forces are then evaluated in parallel with very few communications compared to a purely spatial decomposition.

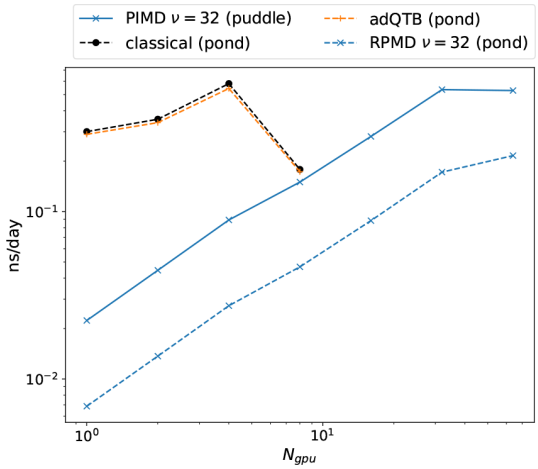


Figure 5. Scaling tests on multi-GPU architecture for the different methods with the AMOEBA polarizable force field. Performance is indicated by the number of nanoseconds of simulation per walltime day as a function of the number of processes. Nodes are composed of four interconnected V100 GPUs so that when using more than four GPUs, out-of-node communications are required, causing a drop in the efficiency.

4. PERSPECTIVE 1: INCLUSION OF QUANTUM NUCLEAR EFFECTS IN MACHINE LEARNING POTENTIALS SIMULATIONS USING THE DEEP-HP PLATFORM

Our platform for nuclear quantum effects is fully compatible with the recently developed Deep-HP module[55] of Tinker-HP that enables the use of machine learning potentials (MLPs), such as ANI[10,80] or DeePMD,[9,81] to perform molecular dynamics simulations. It also enables hybrid machine

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

learning/physical force field calculations in a QM/MM-like embedding framework. MLPs in principle require the explicit inclusion of nuclear quantum effects to achieve their best accuracy on thermodynamical properties since they usually are fitted solely on *ab initio* data. It is thus of the utmost importance for future developments of MLPs to be able to efficiently perform quantum MD in order to assess their accuracy. Since the computational cost of MLPs, as of today, is about an order of magnitude greater than that of polarizable force fields such as AMOEBA, coupling them with path integrals requires a lot of computational resources (especially since integration tricks such as multi-timestepping or RPC cannot usually be used for these potentials). The adQTB, on the other hand, provides a much cheaper alternative that allows one to quickly compute thermodynamical properties with good accuracy, as demonstrated in previous literature[37] and as we show in Section 5.

In this section, we show the compatibility of Quantum-HP and Deep-HP by computing radial distribution functions (RDFs) of liquid water using the DeePMD potential. We performed 500 ps of NVT simulation (at experimental density) for a cubic box of 1000 water molecules for both classical and adQTB MD and a smaller box of 216 molecules for PIMD (with $\nu = 32$ beads). Figure 6 shows the oxygen−oxygen RDF

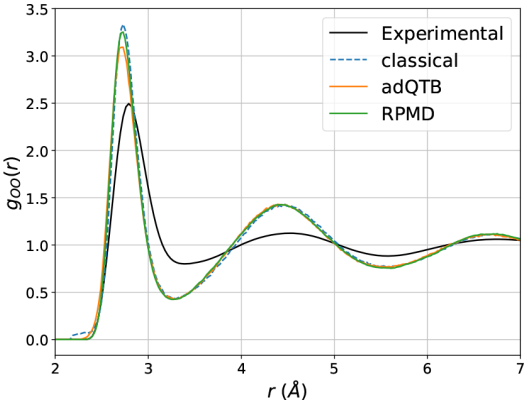


Figure 6. Oxygen−oxygen radial distribution function of water at 300 K computed using the DeePMD ML force field and simulated with classical dynamics (dashed), adQTB (solid orange), and RPMD (solid green). Experimental results from ref.[82]

of liquid water simulated with adQTB, RPMD, and classical MD compared with experimental data from ref 82. The DeePMD model was trained on path integral *ab initio* molecular dynamics (PI-AIMD) trajectories, at the PBE0-TS level (refs 83 and 84): (1) 100,000 snapshots of PI-AIMD liquid water (192 atoms) at 1 bar and 300 K, (2) 20,000 snapshots of PI-AIMD ice phase Ih (288 atoms) at 1 bar and 273 K, (3) 10,000 snapshots of classical AIMD ice phase Ih at 1 bar and 330 K, and (4) 10,000 snapshots of classical AIMD ice phase Ih at 2130 bar and 238 K. We used 10% of the data as the validation set. The DeePMD model was trained using the DeePMD-kit package.[9] The DeePMD model architecture is composed of a (25, 50, 100) embedding net with a 18 neuron-size embedding submatrix and a (240, 240, 120, 60, 30, 10) fitting net. The cutoff radius was set to 6 Å with a smoothing cutoff of 0.5 Å and a two-body embedding descriptor. The final model is trained with 1.2×10^7 Adam steps. With this training setup, the dynamics was stable, and the radial distribution function is in acceptable agreement with experimental results. We note that NQEs appear to be nearly negligible on Figure 6. This can be explained by an almost perfect compensation between competing NQEs:[85−87] the zero-point energies of bending and stretching modes tend to affect the hydrogen bond strength in opposite ways, but the net effect on the structure of the liquid is very small for this particular water model. This net effect is indeed strongly model dependent and can sometimes attenuate the structure of the liquid as in Q-AMOEBA[41] or reinforce it as in MB-Pol.[19] NQEs are more noticeable on the O−H and H−H RDFs (provided in the Supporting Information), especially for peaks corresponding to intramolecular distances which display a strong broadening due to large zero-point energy effects. Since the use of neural networks are the focus of several of our further works, we limit ourselves here concerning the tests but we can already conclude that the Quantum-HP platform can now be used together with the Deep-HP module to efficiently fuel deep neural network simulations including explicit NQEs.

5. PERSPECTIVE 2: INCLUSION OF QUANTUM NUCLEAR EFFECTS IN POLARIZABLE SIMULATIONS: APPLICATION TO HYDRATION FREE ENERGIES OF SMALL ORGANIC MOLECULES

In this last section, we illustrate the capabilities of the platform by computing hydration free energies (HFE) of small organic molecules using the adQTB method. We demonstrate state-of-the-art accuracy with the recently developed Q-AMOEBA water potential for the solvent and Poltype parametrization of the solutes[88] on a benchmark of 40 of the most common organic molecules.[49,89]

Let us first focus on the estimation of free energy differences within quantum simulations. For this study, we used the Bennett Acceptance Ratio (BAR) method[90] that can readily be generalized to the path-integral formalism. Let us denote V_A and V_B the potential energies of two thermodynamical states. The free energy difference between the two states is defined as $\Delta F_{AB} = \beta^{-1} \ln(Z_A/Z_B) \approx \beta^{-1} \ln(Z_{A,\nu}/Z_{B,\nu})$ with Z_A and Z_B the quantum partition functions of states A and B and their respective path-integral counterparts $Z_{A,\nu}$ and $Z_{B,\nu}$ (note that we recover the equality in the $\nu \to \infty$ limit). The Path Integral Bennett Acceptance Ratio (PI-BAR) estimator of the free energy difference is then given by

$$\Delta F_{AB} = C + \beta^{-1} \ln \frac{\langle f_\beta(U_{A,\nu} - U_{B,\nu} + C)\rangle_{B,\nu}}{\langle f_\beta(U_{B,\nu} - U_{A,\nu} - C)\rangle_{A,\nu}} \tag{27}$$

$$C = \Delta F_{AB} + \beta^{-1} \ln(n_B/n_A) \tag{28}$$

with $f_\beta(x) = (1 + \exp(\beta x))^{-1}$, $U_{A,\nu}$ and $U_{B,\nu}$ defined as in eq 3, and n_B and n_A the sample sizes used to estimate the corresponding averages. Note that eqs 27 and 28 form a self-consistent set of equations that is solved iteratively.

Although eq 27 is the minimal expected variance estimator for ΔF_{AB},[90] its accuracy still relies on a somewhat large overlap between the probability distributions of states A and B. Thus, direct estimation of hydration free energies (defining state A as the molecule in solution and state B as the gas phase) is in general impossible.[91] In line with standard procedures,[92,93] we compute the hydration free energy as a sum of free energy differences between neighboring states in a thermodynamical path that progressively decouples the solute from the solvent. First, the electrostatic and polarization interactions between

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

the solute and the solvent are turned off by progressively scaling down the permanent multipoles and polarizabilities of the solute. Then, the van der Waals interactions between the solute and the solvent are scaled down to zero (while using a soft-core potential[47,94]). To recover the hydration free energies, the solute is then "recharged" in the gas phase (i.e., the intramolecular electrostatic interactions are turned back on progressively).

To check the consistency of the methods and the accuracy of the Q-AMOEBA water potential, we first computed the solvation free energy of a Q-AMOEBA water molecule in Q-AMOEBA water. We used a progressive decoupling with 20 thermodynamic states (the precise decoupling schedule that we used is provided in the Supporting Information) that were all simulated using a BAOAB-RESPA integrator with an inner time step of 0.2 fs and an outer time step of 2 fs in the NVT ensemble at 300 K and experimental density. For each thermodynamical state, we thermalize the system for 1 ns and accumulate statistics for 3 ns. The PI-BAR method yields a free energy difference of −5.70 ± 0.05 kcal/mol while the classical BAR value is −6.44 ± 0.04 kcal/mol, demonstrating the strong influence of nuclear quantum effects on the HFE. We note that, for the original AMOEBA force field in classical MD, the HFE was previously reported at −5.86 ± 0.19.[95] Thus, as could be expected, the Q-AMOEBA results with explicit NQEs are close to that of classical simulations with AMOEBA, which was fitted in such a way that it implicitly includes NQEs. On the other hand, the experimental HFE for water was measured at −6.32 kcal/mol. The underestimation of the absolute value of the HFE by Q-AMOEBA is consistent with previous results reported for the enthalpy of vaporization (underestimated by approximately 1 kcal/mol[41]) and the general interpretation that Q-AMOEBA slightly underestimates the strength of hydrogen bonds. We also performed path integrals simulations with a two-stage contraction scheme (with long-range forces and polarization estimated on the centroid only, nonbonded short-range forces evaluated on 12 beads, and bonded forces on the whole 32 beads polymer) and obtained a HFE of −5.84 ± 0.05 kcal/mol, in good agreement with the complete 32 beads calculation, while saving a factor ∼6 on computation time.

While an unbiased estimator of free energy differences can analytically be derived from the path-integral partition function (eq 27), this is not the case for the adaptive QTB. Previous work, however, showed that the probability distribution sampled by the adQTB is usually very close to that of a single bead of the ring polymer (i.e., the correct quantum distribution) such that the estimation of configurational averages with the adQTB is in general accurate. Equation 27, however, is peculiar as it involves the average value of a nonlinear function of the bead-averaged potentials $U_{A,\nu}$, $U_{B,\nu}$. In principle, it could therefore be affected by instantaneous correlations between the beads that the adQTB cannot capture. In practice, however, one can show (see the Appendix) that the bias induced by replacing the bead-averaged potential in eq 27 by the value of the potential on a single bead is of order at least two in the potential energy difference $V_A - V_B$ (i.e., negligible when the decoupling is sufficiently gradual). Indeed, we verified numerically that in the case of Q-AMOEBA water the single-bead HFE is statistically indistinguishable from the unbiased estimator (see Figure 7). Thus, correlations between beads seem only to play a minor role in the free energy estimation, and in turn, the adQTB

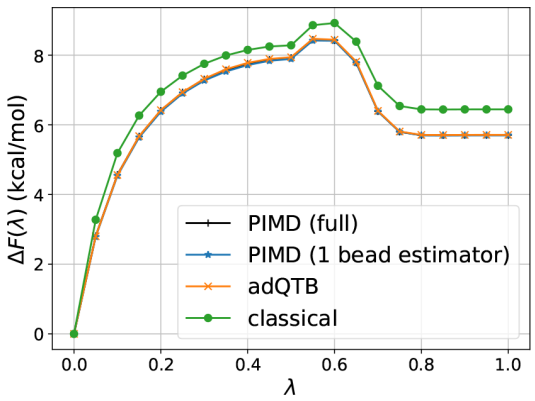


Figure 7. Q-AMOEBA potential of mean force along the vaporization thermodynamical path. $\lambda = 0$ corresponds to the fully solvated molecule, and $\lambda = 1$ corresponds to the noninteracting solute/solvent.

should provide accurate free energy differences using the standard BAR estimator. Indeed, the HFE for water computed using the adQTB is −5.71 ± 0.04 kcal/mol, in excellent agreement with the full path-integrals calculation. Figure 7 shows the potential of mean force along the thermodynamical path and demonstrates that the accuracy of the adQTB (and of the single bead estimator) is not due to error compensations along the path and that the free energy difference at each window is indeed accurately estimated.

We then proceeded to compute the hydration free energies for a benchmark of approximately 40 small organic molecules. All simulations were performed using the same setup as for the calculations on water. We used Q-AMOEBA to model the solvent, and the solutes were parametrized using the Poltype2 software,[88] except for methylamine and dimethylamine for which AMOEBA09 parameters[49] were used. Figure 8 shows a

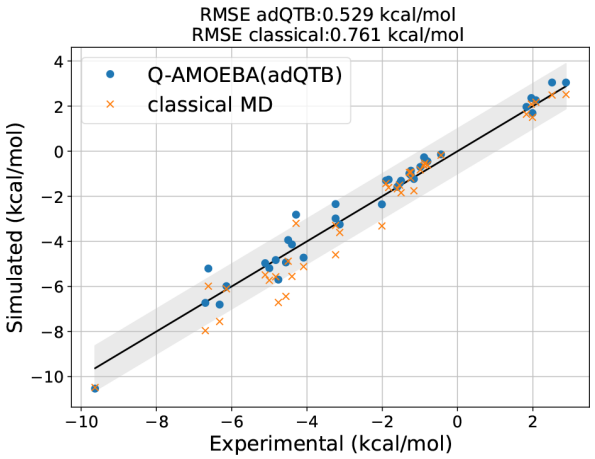


Figure 8. Q-AMOEBA hydration free energies of small organic molecules simulated with adQTB and classical MD compared to experimental values (experimental data and molecules parametrization from refs 49 and 88).

scatter plot of the adQTB and classical HFE against experimental values. We clearly see as a systematic trend that nuclear quantum effects tend to hinder solvation, which is likely due to a weakening of hydrogen bonding between the solute and solvent when including NQEs. Including NQEs also brings the results closer to the experimental values, with a correlation coefficient r^2 of 0.97 and a root-mean-square error

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

(RMSE) over the data set of 0.52 kcal/mol using the adQTB compared to r^2 = 0.93 and RMSE of 0.76 kcal/mol when neglecting NQEs. We also note that our results are in slightly better agreement with the experimental data than those previously reported over the same solutes data set, but without explicit inclusion of NQEs (RMSE of 0.58 kcal/mol[88] using the original AMOEBA parametrization for the solvent that implicitly includes NQEs). Importantly, this improvement has been obtained without fitting the parameters of the force field on the experimental HFEs, thus reinforcing the idea that explicitly taking into account NQEs allows for the development of more transferable models.

6. CONCLUSIONS

We introduced a new platform inside Tinker-HP that enables the explicit inclusion of nuclear quantum effects (NQEs) in molecular dynamics (MD) simulations. The platform, denoted Quantum-HP, implements two methods for quantum MD: ring-polymer MD and adaptive quantum thermal bath MD. While the former provides exact reference results at a relatively high computational cost, the latter was previously shown to give a reliable approximation of NQEs[37] and was the method of choice for the development of the new Q-AMOEBA force field.[41]

The Quantum-HP platform is massively parallel and supports multi-GPU architectures. We have shown that the cost and scaling of the adQTB method is almost identical to that of classical MD and that path integrals, although more expensive, display excellent scaling up to thousands of CPUs and hundreds of GPUs thanks to a two-level parallelization scheme. This makes path-integral MD on Tinker-HP a good candidate to be able to harness the computational power of exascale machines for simulations with new generation models.

We demonstrated the applicability of our platform for the computation of the hydration free energy of small ligands. In these simulations, the solvent was modeled using the newly introduced Q-AMOEBA[41] potential while the ligands were parametrized using the Poltype2[88] software. We showed that the explicit inclusion of NQEs improves the accuracy of such free energies so that future models should be designed with this knowledge, while the additional cost is still affordable when using the adQTB.

The efficiency and massive parallelization capabilities of the newly introduced platform now allows the inclusion of explicit nuclear quantum effects in very large systems. This should be particularly relevant for simulations in extreme conditions of pressure or temperature where NQEs can be massive[20,22,96,97] or to investigate disorder effects in solids for which NQEs can be determinant and large supercells required.[98−100] Importantly, being able to simulate quantum nuclei enables the study of isotope effects that are simply not reachable using classical MD.[101,102] Finally, it opens up the possibility of investigating quantitatively the importance of NQEs in biological processes[38,39] and the subtleties of hydrogen-bonded systems.[87] While the methods for quantum MD are now readily available in Tinker-HP, it will be necessary to reparametrize some of the force fields to avoid double counting of implicit and explicit NQEs, as was shown in the case of Q-AMOEBA for water.[41] The parametrization of Q-AMOEBA for ions, organic molecules, and biomolecules will thus be at the forefront of near future developments. In addition, it will also enable explicit NQEs simulations with advanced polarizable potentials. Models natively designed to reproduce the Born−Oppenheimer surface such as SIBFA[53,54] should directly be applicable, whereas a reparametrization of approaches such as AMOEBA+[51] or HIPPO[103] will be necessary. Neural networks such as ANI,[10] DeePMD,[9] and Physnet[104] (and others) can also be used directly with Quantum-HP. Therefore, with the improvements in computing power and the availability of the methods, we expect that explicit NQEs will soon be an integral part of the standard workflow for both the force field developer and the MD practitioner. Furthermore, the platform will be naturally extended to methods that rely on the simultaneous simulation of multiple replicas of one system such as replica exchange,[15] adaptive sampling,[105] or adaptive bias methods using multiple walkers.[106]

APPENDIX: SINGLE-BEAD PI-BAR ESTIMATOR

We define the single-bead PI-BAR estimator as

$$\Delta F_{AB}^{(1\text{bead})} = C + \beta^{-1} \ln \frac{\langle f_\beta(V_A - V_B + C)\rangle_{B,\nu}}{\langle f_\beta(V_B - V_A - C)\rangle_{A,\nu}} \tag{29}$$

where $V_A = V_A(x_0^{(\nu)}(Q))$ is the potential energy of state A estimated at the position of a single bead of the ring polymer (the choice of the bead index is arbitrary thanks to the cyclic permutation invariance of the ring polymer). In the following, we show that this estimator is biased with respect to eq 27 only to second order (at least) in the difference $\Delta V = V_A - V_B$. For this, let us denote r_{AB} the ratio of average values in eq 29 and write it in terms of explicit integrals over the corresponding distributions:

$$r_{AB} = \frac{Z_{A,\nu}}{Z_{B,\nu}} \frac{\int \mathrm{d}Q f_\beta(\Delta V + C) e^{-\beta(U_{B,\nu}+K)}}{\int \mathrm{d}Q f_\beta(-\Delta V - C) e^{-\beta(U_{A,\nu}+K)}} \tag{30}$$

with $K = \sum_{n>0} 1/2\omega_n^2 Q_n^T M Q_n$ the harmonic potential of the ring polymer. We now use the property of the Fermi function $f_\beta(x) = f_\beta(-x)e^{-\beta x}$ to obtain

$$r_{AB} = \frac{Z_{A,\nu} e^{-\beta C}}{Z_{B,\nu}} \times \frac{\int \mathrm{d}Q f_\beta(-\Delta V - C) e^{-\beta(U_{A,\nu}+K)} e^{-\beta(\Delta V - \Delta U_\nu)}}{\int \mathrm{d}Q f_\beta(-\Delta V - C) e^{-\beta(U_{A,\nu}+K)}} \tag{31}$$

with $\Delta U_\nu = U_{A,\nu} - U_{B,\nu} = \sum_{i=0}^{\nu} \Delta V(x_i^{(\nu)}(Q))/\nu$ the average potential energy difference over the beads of the ring polymer. Expanding the term $e^{-\beta(\Delta V - \Delta U_\nu)}$ in the numerator then gives

$$r_{AB} = \frac{Z_{A,\nu} e^{-\beta C}}{Z_{B,\nu}} \left(1 - \beta \frac{\langle f_\beta(-\Delta V - C)(\Delta V - \Delta U_\nu)\rangle_{A,\nu}}{\langle f_\beta(-\Delta V - C)\rangle_{A,\nu}} + O(\Delta V^2)\right) \tag{32}$$

The only contribution to the first order comes from the order zero in the expansion of the Fermi function which is $f_\beta(-\Delta V - C) = f_\beta(-C) + O(\Delta V)$. Since C is a constant in the integration over Q, we obtain

$$r_{AB} = \frac{Z_{A,\nu} e^{-\beta C}}{Z_{B,\nu}} (1 - \beta(\langle \Delta V\rangle_{A,\nu} - \langle \Delta U_\nu\rangle_{A,\nu}) + O(\Delta V^2)) \tag{33}$$

Due to the cyclic permutation invariance of the ring polymer, we have $\langle \Delta V\rangle_{A,\nu} = \langle \Delta U_\nu\rangle_{A,\nu}$ so that the first order cancels out. Plugging back r_{AB} in the single-bead estimator (eq

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

29), we then see that $\Delta F_{AB}^{(1\text{bead})}$ is unbiased at least up to second order in ΔV.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.2c01233.

Decoupling schedules used for estimating hydration free energies presented in the main text and values and standard errors of hydration free energies for the data set of small organic molecules and the three radial distribution functions of liquid water computed using the DeePMD neural network potential (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Thomas Plé − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France*; Email: thomas.ple@sorbonne-universite.fr

Louis Lagardère − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France*; Email: louis.lagardere@sorbonne-universite.fr

Jean-Philip Piquemal − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France; Institut Universitaire de France, 75005 Paris, France; Department of Biomedical Engineering, The University of Texas at Austin, Austin, Texas 78712, United States;* orcid.org/0000-0001-6615-9426; Email: jean-philip.piquemal@sorbonne-universite.fr

Authors

Nastasia Mauger − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France*

Olivier Adjoua − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France*

Théo Jaffrelot Inizan − *Sorbonne Université, LCT, UMR 7616 CNRS, F-75005 Paris, France*

Simon Huppert − *Institut des Nanosciences de Paris (INSP), CNRS UMR 7588, and Sorbonne Université, F-75005 Paris, France*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.2c01233

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 810367), project EMC2 (J.-P.P.). Computations have been performed at GENCI on the Jean-Zay machine (IDRIS, Orsay, France) and on the Joliot-Curie cluster (TGCC, Bruyères le Châtel, France) on Grant No. A0070707671.

■ REFERENCES

(1) Case, D. A.; Cheatham, T. E., III; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M., Jr; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668−1688.

(2) Brooks, B. R.; Brooks, C. L., III; Mackerell, A. D., Jr.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; Caflisch, A.; Caves, L.; Cui, Q.; Dinner, A. R.; Feig, M.; Fischer, S.; Gao, J.; Hodoscek, M.; Im, W.; Kuczera, K.; Lazaridis, T.; Ma, J.; Ovchinnikov, V.; Paci, E.; Pastor, R. W.; Post, C. B.; Pu, J. Z.; Schaefer, M.; Tidor, B.; Venable, R. M.; Woodcock, H. L.; Wu, X.; Yang, W.; York, D. M.; Karplus, M. CHARMM: the biomolecular simulation program. *J. Comput. Chem.* **2009**, *30*, 1545−1614.

(3) Van Der Spoel, D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. GROMACS: fast, flexible, and free. *J. Comput. Chem.* **2005**, *26*, 1701−1718.

(4) Shi, Y.; Ren, P.; Schnieders, M.; Piquemal, J.-P. *Reviews in Computational Chemistry*, Vol. *28*; John Wiley & Sons, Ltd, 2015; Chapter 2, pp 51−86.

(5) Reddy, S. K.; Straight, S. C.; Bajaj, P.; Huy Pham, C.; Riera, M.; Moberg, D. R.; Morales, M. A.; Knight, C.; Götz, A. W.; Paesani, F. On the accuracy of the MB-pol many-body potential for water: Interaction energies, vibrational frequencies, and classical thermodynamic and dynamical properties from clusters to liquid water and ice. *J. Chem. Phys.* **2016**, *145*, 194504.

(6) Melcr, J.; Piquemal, J.-P. Accurate biomolecular simulations account for electronic polarization. *J. Melcr, J.-P. Piquemal, Front. Mol. Biosci.* **2019**, *6*, 143.

(7) Jing, Z.; Liu, C.; Cheng, S. Y.; Qi, R.; Walker, B. D.; Piquemal, J.-P.; Ren, P. Polarizable Force Fields for Biomolecular Simulations: Recent Advances and Applications. *Annu. Rev. Biophys.* **2019**, *48*, 371−394.

(8) Behler, J.; Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.

(9) Zhang, L.; Han, J.; Wang, H.; Car, R.; Weinan, E. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. *Phys. Rev. Lett.* **2018**, *120*, 143001.

(10) Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.* **2017**, *8*, 3192−3203.

(11) Tuckerman, M.; Berne, B. J.; Martyna, G. J. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.* **1992**, *97*, 1990−2001.

(12) Lagardère, L.; Aviat, F.; Piquemal, J.-P. Pushing the limits of multiple-time-step strategies for polarizable point dipole molecular dynamics. *J. Phys. Chem. Lett.* **2019**, *10*, 2593−2599.

(13) Duane, S.; Kennedy, A. D.; Pendleton, B. J.; Roweth, D. Hybrid monte carlo. *Phys. Lett. B* **1987**, *195*, 216−222.

(14) Girolami, M.; Calderhead, B. Riemann manifold langevin and hamiltonian monte carlo methods. *J. R. Stat. Soc., B: Stat. Methodol.* **2011**, *73*, 123−214.

(15) Sugita, Y.; Okamoto, Y. Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.* **1999**, *314*, 141−151.

(16) Laio, A.; Parrinello, M. Escaping free-energy minima. *Proc. Natl. Acad. Sci. U.S.A.* **2002**, *99*, 12562−12566.

(17) Paesani, F.; Iuchi, S.; Voth, G. A. Quantum effects in liquid water from an ab initio-based polarizable force field. *J. Chem. Phys.* **2007**, *127*, 074506.

(18) Fanourgakis, G. S.; Schenter, G. K.; Xantheas, S. S. A quantitative account of quantum effects in liquid water. *J. Chem. Phys.* **2006**, *125*, 141102.

(19) Li, C.; Paesani, F.; Voth, G. A. Static and Dynamic Correlations in Water: Comparison of Classical Ab Initio Molecular Dynamics at Elevated Temperature with Path Integral Simulations at Ambient Temperature. *J. Chem. Theory Comput.* **2022**, *18*, 2124−2131.

(20) Benoit, M.; Marx, D.; Parrinello, M. Tunnelling and zero-point motion in high-pressure ice. *Nature* **1998**, *392*, 258−261.

(21) Hirshberg, B.; Rizzi, V.; Parrinello, M. Path integral molecular dynamics for bosons. *Proc. Natl. Acad. Sci. U.S.A.* **2019**, *116*, 21445−21449.

(22) Schaack, S.; Depondt, P.; Huppert, S.; Finocchi, F. Quantum driven proton diffusion in brucite-like minerals under high pressure. *Sci. Rep.* **2020**, *10*, 1−10.

(23) Brieuc, F.; Schran, C.; Uhl, F.; Forbert, H.; Marx, D. Converged quantum simulations of reactive solutes in superfluid helium: The Bochum perspective. *J. Chem. Phys.* **2020**, *152*, 210901.

(24) Myung, C. W.; Hirshberg, B.; Parrinello, M. Prediction of a supersolid phase in high-pressure deuterium. *Phys. Rev. Lett.* **2022**, *128*, 045301.
(25) Paesani, F.; Voth, G. A. The properties of water: Insights from quantum simulations. *J. Phys. Chem. B* **2009**, *113*, 5702−5719.
(26) Schwartz, C. P.; Uejio, J. S.; Saykally, R. J.; Prendergast, D. On the importance of nuclear quantum motions in near edge x-ray absorption fine structure spectroscopy of molecules. *J. Chem. Phys.* **2009**, *130*, 184109.
(27) Ceriotti, M.; Cuny, J.; Parrinello, M.; Manolopoulos, D. E. Nuclear quantum effects and hydrogen bond fluctuations in water. *Proc. Natl. Acad. Sci. U.S.A.* **2013**, *110*, 15591−15596.
(28) Engel, E. A.; Kapil, V.; Ceriotti, M. Importance of nuclear quantum effects for NMR crystallography. *J. Phys. Chem. Lett.* **2021**, *12*, 7701−7707.
(29) Feynman, R. P.; Hibbs, A. R.; Styer, D. F. *Quantum Mechanics and Path Integrals*; Courier Corporation, 2010.
(30) Berne, B. J.; Thirumalai, D. On the simulation of quantum systems: path integral methods. *Annu. Rev. Phys. Chem.* **1986**, *37*, 401−424.
(31) Ceriotti, M.; Bussi, G.; Parrinello, M. Nuclear Quantum Effects in Solids Using a Colored-Noise Thermostat. *Phys. Rev. Lett.* **2009**, *103*, 030603.
(32) Dammak, H.; Chalopin, Y.; Laroche, M.; Hayoun, M.; Greffet, J.-J. Quantum thermal bath for molecular dynamics simulation. *Phys. Rev. Lett.* **2009**, *103*, 190601.
(33) Ceriotti, M.; Manolopoulos, D. E.; Parrinello, M. Accelerating the convergence of path integral dynamics with a generalized Langevin equation. *J. Chem. Phys.* **2011**, *134*, 084104.
(34) Brieuc, F.; Dammak, H.; Hayoun, M. Quantum thermal bath for path integral molecular dynamics simulation. *J. Chem. Theory Comput.* **2016**, *12*, 1351−1359.
(35) Mangaud, E.; Huppert, S.; Plé, T.; Depondt, P.; Bonella, S.; Finocchi, F. The Fluctuation−Dissipation Theorem as a Diagnosis and Cure for Zero-Point Energy Leakage in Quantum Thermal Bath Simulations. *J. Chem. Theory Comput.* **2019**, *15*, 2863−2880.
(36) Huppert, S.; Plé, T.; Bonella, S.; Depondt, P.; Finocchi, F. Simulation of Nuclear Quantum Effects in Condensed Matter Systems via Quantum Baths. *Appl. Sci.* **2022**, *12*, 4756.
(37) Mauger, N.; Plé, T.; Lagardère, L.; Bonella, S.; Mangaud, É.; Piquemal, J.-P.; Huppert, S. Nuclear Quantum Effects in liquid water at near classical computational cost using the adaptive Quantum Thermal Bath. *J. Phys. Chem. Lett.* **2021**, *12*, 8285−8291.
(38) Fang, W.; Chen, J.; Rossi, M.; Feng, Y.; Li, X.-Z.; Michaelides, A. Inverse temperature dependence of nuclear quantum effects in dna base pairs. *J. Phys. Chem. Lett.* **2016**, *7*, 2125−2131.
(39) Law, Y.; Hassanali, A. The importance of nuclear quantum effects in spectral line broadening of optical spectra and electrostatic properties in aromatic chromophores. *J. Chem. Phys.* **2018**, *148*, 102331.
(40) Pereyaslavets, L.; Kurnikov, I.; Kamath, G.; Butin, O.; Illarionov, A.; Leontyev, I.; Olevanov, M.; Levitt, M.; Kornberg, R. D.; Fain, B. On the importance of accounting for nuclear quantum effects in ab initio calibrated force fields in biological simulations. *Proc. Natl. Acad. Sci. U.S.A.* **2018**, *115*, 8878−8882.
(41) Mauger, N.; Plé, T.; Lagardère, L.; Huppert, S.; Piquemal, J.-P. Improving condensed-phase water dynamics with explicit nuclear quantum effects: The polarizable Q-AMOEBA force field. *J. Phys. Chem. B* **2022**, *126*, 8813−8826.
(42) Kapil, V.; Rossi, M.; Marsalek, O.; Petraglia, R.; Litman, Y.; Spura, T.; Cheng, B.; Cuzzocrea, A.; Meißner, R. H.; Wilkins, D. M.; Helfrecht, B. A.; Juda, P.; Bienvenue, S. P.; Fang, W.; Kessler, J.; Poltavsky, I.; Vandenbrande, S.; Wieme, J.; Corminboeuf, C.; Kühne, T. D.; Manolopoulos, D. E.; Markland, T. E.; Richardson, J. O.; Tkatchenko, A.; Tribello, G. A.; Van Speybroeck, V.; Ceriotti, M. i-PI 2.0: A universal force engine for advanced molecular simulations. *Comput. Phys. Commun.* **2019**, *236*, 214−223.
(43) Thompson, A. P.; Aktulga, H. M.; Berger, R.; Bolintineanu, D. S.; Brown, W. M.; Crozier, P. S.; in 't Veld, P. J.; Kohlmeyer, A.; Moore, S. G.; Nguyen, T. D.; Shan, R.; Stevens, M. J.; Tranchida, J.; Trott, C.; Plimpton, S. J. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput. Phys. Commun.* **2022**, *271*, 108171.
(44) Markland, T. E.; Manolopoulos, D. E. An efficient ring polymer contraction scheme for imaginary time path integral simulations. *J. Chem. Phys.* **2008**, *129*, 024105.
(45) Markland, T. E.; Manolopoulos, D. E. A refined ring polymer contraction scheme for systems with electrostatic interactions. *Chem. Phys. Lett.* **2008**, *464*, 256−261.
(46) Ren, P.; Ponder, J. W. Polarizable atomic multipole water model for molecular mechanics simulation. *J. Phys. Chem. B* **2003**, *107*, 5933−5947.
(47) Lagardère, L.; Jolly, L.-H.; Lipparini, F.; Aviat, F.; Stamm, B.; Jing, Z. F.; Harger, M.; Torabifard, H.; Cisneros, G. A.; Schnieders, M. J.; Gresh, N.; Maday, Y.; Ren, P. Y.; Ponder, J. W.; Piquemal, J.-P. Tinker-HP: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields. *Chem. Sci.* **2018**, *9*, 956−972.
(48) Adjoua, O.; Lagardère, L.; Jolly, L.-H.; Durocher, A.; Very, T.; Dupays, I.; Wang, Z.; Inizan, T. J.; Célerse, F.; Ren, P.; Ponder, J. W.; Piquemal, J.-P. Tinker-HP: Accelerating molecular dynamics simulations of large complex systems with advanced point dipole polarizable force fields using GPUs and multi-GPU systems. *J. Chem. Theory Comput.* **2021**, *17*, 2034−2053.
(49) Ren, P.; Wu, C.; Ponder, J. W. Polarizable atomic multipole-based molecular mechanics for organic molecules. *J. Chem. Theory Comput.* **2011**, *7*, 3143−3161.
(50) Shi, Y.; Xia, Z.; Zhang, J.; Best, R.; Wu, C.; Ponder, J. W.; Ren, P. Polarizable atomic multipole-based AMOEBA force field for proteins. *J. Chem. Theory Comput.* **2013**, *9*, 4046−4063.
(51) Liu, C.; Piquemal, J.-P.; Ren, P. AMOEBA+ classical potential for modeling molecular interactions. *J. Chem. Theory Comput.* **2019**, *15*, 4122−4139.
(52) Liu, C.; Piquemal, J.-P.; Ren, P. Implementation of geometry-dependent charge flux into the polarizable AMOEBA+ potential. *J. Phys. Chem. Lett.* **2020**, *11*, 419−426.
(53) Gresh, N.; Cisneros, G. A.; Darden, T. A.; Piquemal, J.-P. Anisotropic, polarizable molecular mechanics studies of inter-and intramolecular interactions and ligand- macromolecule complexes. A bottom-up strategy. *J. Chem. Theory Comput.* **2007**, *3*, 1960−1986.
(54) Naseem-Khan, S.; Lagardère, L.; Narth, C.; Cisneros, G. A.; Ren, P.; Gresh, N.; Piquemal, J.-P. Development of the Quantum-Inspired SIBFA Many-Body Polarizable Force Field: Enabling Condensed-Phase Molecular Dynamics Simulations. *J. Chem. Theory Comput.* **2022**, *18*, 3607−3621.
(55) Inizan, T. J.; Plé, T.; Adjoua, O.; Ren, P.; Gökcan, H.; Isayev, O.; Lagardère, L.; Piquemal, J.-P. Scalable Hybrid Deep Neural Networks/Polarizable Potentials Biomolecular Simulations including long-range effects. *arXiv Preprint*, arXiv:2207.14276, 2022.
(56) Craig, I. R.; Manolopoulos, D. E. Quantum statistics and classical mechanics: Real time correlation functions from ring polymer molecular dynamics. *J. Chem. Phys.* **2004**, *121*, 3368−3373.
(57) Habershon, S.; Manolopoulos, D. E.; Markland, T. E.; Miller, T. F., III Ring-polymer molecular dynamics: Quantum effects in chemical dynamics from classical trajectories in an extended phase space. *Annu. Rev. Phys. Chem.* **2013**, *64*, 387−413.
(58) Hele, T. J. Thermal quantum time-correlation functions from classical-like dynamics. *Mol. Phys.* **2017**, *115*, 1435−1462.
(59) Althorpe, S. C. Path-integral approximations to quantum dynamics. *Eur. Phys. J. B* **2021**, *94*, 1−17.
(60) Braams, B. J.; Manolopoulos, D. E. On the short-time limit of ring polymer molecular dynamics. *J. Chem. Phys.* **2006**, *125*, 124105.
(61) Hele, T. J. H.; Willatt, M. J.; Muolo, A.; Althorpe, S. C. Communication: Relation of centroid molecular dynamics and ring-polymer molecular dynamics to exact quantum dynamics. *J. Chem. Phys.* **2015**, *142*, 191101.
(62) Witt, A.; Ivanov, S. D.; Shiga, M.; Forbert, H.; Marx, D. On the applicability of centroid and ring polymer path integral molecular

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

dynamics for vibrational spectroscopy. *J. Chem. Phys.* **2009**, *130*, 194510.
(63) Benson, R. L.; Trenins, G.; Althorpe, S. C. Which quantum statistics−classical dynamics method is best for water? *Faraday Discuss.* **2020**, *221*, 350−366.
(64) Kubo, R. The fluctuation-dissipation theorem. *Rep. Prog. Phys.* **1966**, *29*, 255.
(65) Basire, M.; Borgis, D.; Vuilleumier, R. Computing Wigner distributions and time correlation functions using the quantum thermal bath method: application to proton transfer spectroscopy. *Phys. Chem. Chem. Phys.* **2013**, *15*, 12591.
(66) Brieuc, F.; Bronstein, Y.; Dammak, H.; Depondt, P.; Finocchi, F.; Hayoun, M. Zero-point energy leakage in Quantum Thermal Bath molecular dynamics simulations. *J. Chem. Theory Comput.* **2016**, *12*, 5688−5697.
(67) Habershon, S.; Manolopoulos, D. E. Zero point energy leakage in condensed phase dynamics: An assessment of quantum simulation methods for liquid water. *J. Chem. Phys.* **2009**, *131*, 244518.
(68) Feller, S. E.; Zhang, Y.; Pastor, R. W.; Brooks, B. R. Constant pressure molecular dynamics simulation: The Langevin piston method. *J. Chem. Phys.* **1995**, *103*, 4613−4621.
(69) Ceriotti, M.; More, J.; Manolopoulos, D. E. i-PI: A Python interface for ab initio path integral molecular dynamics simulations. *Comput. Phys. Commun.* **2014**, *185*, 1019−1026.
(70) Leimkuhler, B.; Matthews, C. Rational construction of stochastic numerical methods for molecular sampling. *Appl. Math. Res. eXpress* **2013**, *2013*, 34−56.
(71) Liu, J.; Li, D.; Liu, X. A simple and accurate algorithm for path integral molecular dynamics with the Langevin thermostat. *J. Chem. Phys.* **2016**, *145*, 024103.
(72) Rossi, M.; Ceriotti, M.; Manolopoulos, D. E. How to remove the spurious resonances from ring polymer molecular dynamics. *J. Chem. Phys.* **2014**, *140*, 234116.
(73) Gillespie, D. T. Exact numerical simulation of the Ornstein-Uhlenbeck process and its integral. *Phys. Rev. E* **1996**, *54*, 2084.
(74) Korol, R.; Rosa-Raíces, J. L.; Bou-Rabee, N.; Miller, T. F., III Dimension-free path-integral molecular dynamics without preconditioning. *J. Chem. Phys.* **2020**, *152*, 104102.
(75) Tuckerman, M. E.; Berne, B. J.; Martyna, G. J.; Klein, M. L. Efficient molecular dynamics and hybrid Monte Carlo algorithms for path integrals. *J. Chem. Phys.* **1993**, *99*, 2796−2808.
(76) Martyna, G. J.; Tuckerman, M. E.; Tobias, D. J.; Klein, M. L. Explicit reversible integrators for extended systems dynamics. *Mol. Phys.* **1996**, *87*, 1117−1157.
(77) Martyna, G. J.; Hughes, A.; Tuckerman, M. E. Molecular dynamics algorithms for path integrals at constant pressure. *J. Chem. Phys.* **1999**, *110*, 3275−3290.
(78) Fanourgakis, G. S.; Markland, T. E.; Manolopoulos, D. E. A fast path integral method for polarizable force fields. *J. Chem. Phys.* **2009**, *131*, 094102.
(79) Marsalek, O.; Markland, T. E. Ab initio molecular dynamics with nuclear quantum effects at classical cost: Ring polymer contraction for density functional theory. *J. Chem. Phys.* **2016**, *144*, 054112.
(80) Devereux, C.; Smith, J. S.; Huddleston, K. K.; Barros, K.; Zubatyuk, R.; Isayev, O.; Roitberg, A. E. Extending the applicability of the ANI deep learning molecular potential to sulfur and halogens. *J. Chem. Theory Comput.* **2020**, *16*, 4192−4202.
(81) Ko, H.-Y.; Zhang, L.; Santra, B.; Wang, H.; E, W.; DiStasio, R. A., Jr; Car, R. Isotope effects in liquid water via deep potential molecular dynamics. *Mol. Phys.* **2019**, *117*, 3269−3281.
(82) Soper, A. K. The radial distribution functions of water as derived from radiation total scattering experiments: Is there anything we can say for sure? *Int. Scholarly Res. Not.* **2013**, *2013*, 1−67.
(83) Ko, H.-Y.; Zhang, L.; Santra, B.; Wang, H.; E, W.; DiStasio, R. A., Jr.; Car, R. Isotope effects in liquid water via deep potential molecular dynamics. *Mol. Phys.* **2019**, *117*, 3269−3281.
(84) Zhang, L.; Wang, H.; Car, R.; E, W. Phase Diagram of a Deep Potential Water Model. *Phys. Rev. Lett.* **2021**, *126*, 236001.
(85) Habershon, S.; Markland, T. E.; Manolopoulos, D. E. Competing quantum effects in the dynamics of a flexible water model. *J. Chem. Phys.* **2009**, *131*, 024501.
(86) Ceriotti, M.; Fang, W.; Kusalik, P. G.; McKenzie, R. H.; Michaelides, A.; Morales, M. A.; Markland, T. E. Nuclear quantum effects in water and aqueous systems: Experiment, theory, and current challenges. *Chem. Rev.* **2016**, *116*, 7529−7550.
(87) Li, X.-Z.; Walker, B.; Michaelides, A. Quantum nature of the hydrogen bond. *Proc. Natl. Acad. Sci. U.S.A.* **2011**, *108*, 6369−6373.
(88) Walker, B.; Liu, C.; Wait, E.; Ren, P. Automation of AMOEBA polarizable force field for small molecules: Poltype 2. *J. Comput. Chem.* **2022**, *43*, 1530−1542.
(89) Wu, J. C.; Chattree, G.; Ren, P. Automation of AMOEBA polarizable force field parameterization for small molecules. *Theor. Chem. Acc.* **2012**, *131*, 1−11.
(90) Bennett, C. H. Efficient estimation of free energy differences from Monte Carlo data. *J. Comput. Phys.* **1976**, *22*, 245−268.
(91) Wu, D.; Kofke, D. A. Phase-space overlap measures. I. Fail-safe bias detection in free energies calculated by molecular simulation. *J. Chem. Phys.* **2005**, *123*, 054103.
(92) Wu, D.; Kofke, D. A. Phase-space overlap measures. II. Design and implementation of staging methods for free-energy calculations. *J. Chem. Phys.* **2005**, *123*, 084109.
(93) Shi, Y.; Wu, C.; Ponder, J. W.; Ren, P. Multipole electrostatics in hydration free energy calculations. *J. Comput. Chem.* **2011**, *32*, 967−977.
(94) Steinbrecher, T.; Joung, I.; Case, D. A. Soft-core potentials in thermodynamic integration: Comparing one-and two-step transformations. *J. Comput. Chem.* **2011**, *32*, 3253−3263.
(95) Ponder, J. W.; Wu, C.; Ren, P.; Pande, V. S.; Chodera, J. D.; Schnieders, M. J.; Haque, I.; Mobley, D. L.; Lambrecht, D. S.; DiStasio, R. A. J.; Head-Gordon, M.; Clark, G. N. I.; Johnson, M. E.; Head-Gordon, T. Current status of the AMOEBA polarizable force field. *J. Phys. Chem. B* **2010**, *114*, 2549−2564.
(96) Markland, T. E.; Ceriotti, M. Nuclear quantum effects enter the mainstream. *Nature Rev. Chem.* **2018**, *2*, 1−14.
(97) Witt, A.; Sebastianelli, F.; Tuckerman, M. E.; Bacic, Z. Path integral molecular dynamics study of small H2 clusters in the large cage of structure II clathrate hydrate: Temperature dependence of quantum spatial distributions. *J. Phys. Chem. C* **2010**, *114*, 20775−20782.
(98) Geneste, G.; Dammak, H.; Hayoun, M.; Thiercelin, M. Low-temperature anharmonicity of barium titanate: A path-integral molecular-dynamics study. *Phys. Rev. B* **2013**, *87*, 014113.
(99) Bronstein, Y.; Depondt, P.; Bove, L. E.; Gaal, R.; Saitta, A. M.; Finocchi, F. Quantum versus classical protons in pure and salty ice under pressure. *Phys. Rev. B* **2016**, *93*, 024104.
(100) Fallacara, E.; Depondt, P.; Huppert, S.; Ceotto, M.; Finocchi, F. Thermal and Nuclear Quantum Effects at the Antiferroelectric to Paraelectric Phase Transition in KOH and KOD Crystals. *J. Phys. Chem. C* **2021**, *125*, 22328−22334.
(101) Ceriotti, M.; Markland, T. E. Efficient methods and practical guidelines for simulating isotope effects. *J. Chem. Phys.* **2013**, *138*, 014112.
(102) Cheng, B.; Ceriotti, M. Direct path integral estimators for isotope fractionation ratios. *J. Chem. Phys.* **2014**, *141*, 244112.
(103) Rackers, J. A.; Silva, R. R.; Wang, Z.; Ponder, J. W. Polarizable water potential derived from a model electron density. *J. Chem. Theory Comput.* **2021**, *17*, 7056−7084.
(104) Unke, O. T.; Meuwly, M. PhysNet: A Neural Network for Predicting Energies, Forces, Dipole Moments, and Partial Charges. *J. Chem. Theory Comput.* **2019**, *15*, 3678−3693.
(105) Jaffrelot Inizan, T.; Célerse, F.; Adjoua, O.; El Ahdab, D.; Jolly, L.-H.; Liu, C.; Ren, P.; Montes, M.; Lagarde, N.; Lagardère, L.; Monmarché, P.; Piquemal, J.-P. High-resolution mining of the SARS-CoV-2 main protease conformational space: supercomputer-driven unsupervised adaptive sampling. *Chem. Sci.* **2021**, *12*, 4889−4907.

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

(106) Minoukadeh, K.; Chipot, C.; Lelièvre, T. Potential of mean force calculations: a multiple-walker adaptive biasing force approach. *J. Chem. Theory Comput.* **2010**, *6*, 1008−1017.

https://doi.org/10.1021/acs.jctc.2c01233
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

Conclusion

The efficiency and extensive parallelization capabilities of Quantum-HP enable the explicit inclusion of NQEs in large-scale systems. This paves the way for quantitative investigations into the importance of NQEs in biological phenomena [165, 166] and hydrogen-bonded systems [167]. Quantum-HP is compatible with popular MLPs such as ANI[99, 101, 102], DeePMD[104, 103], and others, allowing direct utilization of these models. As a result, with the increasing computing power and availability of these methods, it is expected that explicit NQEs will soon become an integral part of the standard workflow for both FF and MLP developers. Furthermore, the platform can naturally be extended to methods that involve the simultaneous simulation of multiple replicas of a system, such as the adaptive sampling algorithm discussed in the next chapter.

2.4 Perspective: Tight integration of Neural Networks in the AMOEBA polarizable Force Field, Q-AMOEBA-NN

2.4.1 Introduction

FFs are an approximation of the true potential energy surface described by quantum mechanics, while ML potentials (MLP) avoid solving the Schrödinger equation by providing a mathematical direct relation between the atomic positions and the potential energy. However, MLPs are limited to modeling short-range interactions making them unsuitable to capture long-range interactions that are essential for the stability of bio-molecules.
To address these challenges, we propose two models NN-AMOEBA and Q-AMOEBA-NN, where neural networks are trained on short-range interactions while long-range effects are treated using the physical AMOEBA model without and with NQEs, Q-AMOEBA-NN.

2.4.2 The NN-AMOEBA model

The NN-AMOEBA total potential energies is defined as:

$$\mathbf{E}_{tot} = \mathbf{E}_{\text{bond}}^{\text{AMOEBA}} + \mathbf{E}_{\text{NN}} + \mathbf{E}_{\text{VdW}}^{\text{AMOEBA}} + \mathbf{E}_{\text{el}}^{\text{AMOEBA}} + \mathbf{E}_{\text{pol}}^{\text{AMOEBA}} \tag{2.1}$$

We retained the intermolecular terms of AMOEBA and the bond-stretching term of the intramolecular potential to prevent bond dissociation and reactivity. Developing a reactive FF is a tremendous task that is beyond the scope of this thesis.
Indeed, Reactive FFs, such as ReaxFF [168], PhysNet [169], MS-ARMD [170], are parametrized FFs that can describe chemical reactions and bond breaking events that could occur during simulations. These FFs are typically parametrized on relatively small systems, such as small clusters or gas phase molecules, thus limiting their transferability to larger and more complex systems, such as biomolecules.
In particular, the transferability of reactive FFs to biomolecular simulations is not obvious due to several factors. Firstly, the interactions between biomolecules and the solvent, e.g water, complicates the parameterization process which have crucial role in biochemistry, see Chapter 4. Secondly, biomolecules are much larger and are usually more structured that molecules which can result in inadequate parametrization. Finally, the chemical reactions that occur in biomolecules can involve a large number of atoms and are not well known in the literature which complicates reactive FFs as very few experimental data and QM computations exist.
Moreover, the atomic multipoles and VdW parameters of AMOEBA were optimized based on bond length equilibrium distances. Although, the bond-stretching term could be replaced by a flat-bottom potential, this would be equivalent to the already implemented bond-stretching term. Overall, our

NN-AMOEBA approach aims to provide a balance description between short and long-range interactions, making it more easily transferable for large biological systems.

2.4.3 The Q-AMOEBA-NN model

While NN-AMOEBA can be trained on the relative difference between the AMOEBA and QM energies, one remaining issue is the coupling with NQEs, which is not evident. The classical parametrization procedure of AMOEBA with poltype and force balanced, explained earlier in the thesis, incorporate implicitly the NQEs inside the VdW term. Thus, to avoid double counting with NQEs methods, the VdW terms must be also parametrized during the training procedure. From the NN-AMOEBA model we are also adding a neural network which is train on VdW parameters, QM energies and forces. From this, we can reach a fully QM NN-AMOEBA model. Once trained we then combined it with the aQTB model, explained earlier in the thesis. The model accuracy and computational efficiency will be directly compared to the AMOEBA which is already a significant step toward the definition of an accurate FF.

2.4.4 Conclusion

Overall, this Q-AMOEBA-NN model aims to push the limit of FF toward QM accuracy. It has been tested on host-guest binding free energies and protein folding. Powered by the newly developed multi-GPU ML platform, Deep-HP, the models is able to use thousands of GPU cards and therefore to simulate millions of atoms systems. Thanks to the model flexibility, multiple strategies for improvements are possible, e.g: larger data sets, improved neural network architectures and better atomic environment vectors.

3 Unsupervised Data-Driven Enhanced Sampling Techniques, High-Performance Computing, and GPU for Accelerating Large-Scale Simulations

3.1 Unsupervised Data-Driven Adaptive Sampling technique to accelerate conformational space sampling

Introduction

Conventional MD is not able to efficiently explore the conformational space of most systems including proteins. To overcome this limitation, MD are coupled with enhanced sampling techniques, which aim to escape metastable states and accelerate the discovery of previously unseen states. These techniques rely on what we called collective variables (CVs) and was introduced in Chapter 1.5 Section 1.5.1 However, CVs can be challenging to comprehend due to their inherently multibody and emergent nature, particularly in the case of complex protein conformational spaces. Data-driven techniques can help estimate CVs from molecular simulation data in a systematic manner, thereby guiding the exploration of enhanced sampling models. In this section, we propose a density-driven unsupervised parallel-in-time adaptive sampling method that enables multi-microsecond MD simulations of large systems. Combining this technique with the Tinker-HP GPU extension, discussed in the previous section, and the AMOEBA polarizable force fields, allow to speed up the exploration of the conformational space of large systems at high resolution. To extract meaningful macrostates from the resulting large high-dimensional trajectory data, we utilized multiple unsupervised machine learning clustering algorithms. Our results demonstrate that this approach can address various challenges, particularly the pressing modeling problem posed by the SARS-CoV-2 Main Protease, by revealing previously unknown structural behaviors and cryptic pockets, and shedding light on the influence of pH on protein stability.[171]

Chemical Science

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE

View Article Online
View Journal | View Issue

Cite this: *Chem. Sci.*, 2021, **12**, 4889

All publication charges for this article have been paid for by the Royal Society of Chemistry

Received 10th January 2021
Accepted 27th January 2021

DOI: 10.1039/d1sc00145k

rsc.li/chemical-science

High-resolution mining of the SARS-CoV-2 main protease conformational space: supercomputer-driven unsupervised adaptive sampling†

Théo Jaffrelot Inizan, ‡[a] Frédéric Célerse, ‡[ab] Olivier Adjoua,[a] Dina El Ahdab, [ac] Luc-Henri Jolly,[d] Chengwen Liu,[e] Pengyu Ren,[e] Matthieu Montes,[f] Nathalie Lagarde,[f] Louis Lagardère,*[ad] Pierre Monmarché*[ag] and Jean-Philip Piquemal *[aeh]

We provide an unsupervised adaptive sampling strategy capable of producing μs-timescale molecular dynamics (MD) simulations of large biosystems using many-body polarizable force fields (PFFs). The global exploration problem is decomposed into a set of separate MD trajectories that can be restarted within a selective process to achieve sufficient phase-space sampling. Accurate statistical properties can be obtained through reweighting. Within this highly parallel setup, the Tinker-HP package can be powered by an arbitrary large number of GPUs on supercomputers, reducing exploration time from years to days. This approach is used to tackle the urgent modeling problem of the SARS-CoV-2 Main Protease (M^{pro}) producing more than 38 μs of all-atom simulations of its apo (ligand-free) dimer using the high-resolution AMOEBA PFF. The first 15.14 μs simulation (physiological pH) is compared to available non-PFF long-timescale simulation data. A detailed clustering analysis exhibits striking differences between FFs, with AMOEBA showing a richer conformational space. Focusing on key structural markers related to the oxyanion hole stability, we observe an asymmetry between protomers. One of them appears less structured resembling the experimentally inactive monomer for which a 6 μs simulation was performed as a basis for comparison. Results highlight the plasticity of the M^{pro} active site. The C-terminal end of its less structured protomer is shown to oscillate between several states, being able to interact with the other protomer, potentially modulating its activity. Active and distal site volumes are found to be larger in the most active protomer within our AMOEBA simulations compared to non-PFFs as additional cryptic pockets are uncovered. A second 17 μs AMOEBA simulation is performed with protonated His172 residues mimicking lower pH. Data show the protonation impact on the destructuring of the oxyanion loop. We finally analyze the solvation patterns around key histidine residues. The confined AMOEBA polarizable water molecules are able to explore a wide range of dipole moments, going beyond bulk values, leading to a water molecule count consistent with experimental data. Results suggest that the use of PFFs could be critical in drug discovery to accurately model the complexity of the molecular interactions structuring M^{pro}.

1 Introduction

At the end of December 2019, a novel coronavirus (CoV) that induces severe acute respiratory disease (SARS) was discovered and labeled SARS-CoV-2.[1] It causes the disease named COVID-19, which led to a global pandemic in 2020 and finally to an urgent global issue.

Great effort has been made to gain insights into the action of the virus on the human body. As the genome of the virus has been rapidly determined,[2] a similarity between the SARS-CoV-2 virus and the older SARS-CoV (2003) and Middle East respiratory syndrome coronavirus (MERS-CoV in 2012) was observed. Besides vaccines, researchers started the hunt for small molecules to treat the disease. Rapidly,[2] different classes of proteins have been experimentally characterized that could be useful

[a]*Sorbonne Université, LCT, UMR 7616 CNRS, Paris, France. E-mail: louis.lagardere@sorbonne-universite.fr; pierre.monmarche@sorbonne-universite.fr; jean-philip.piquemal@sorbonne-universite.fr*
[b]*Sorbonne Université, IPCM, UMR 8232 CNRS, Paris, France*
[c]*Université Saint-Joseph de Beyrouth, UR-EGP Faculté des Sciences, Lebanon*
[d]*Sorbonne Université, IP2CT, FR 2622 CNRS, Paris, France*
[e]*University of Texas at Austin, Department of Biomedical Engineering, Texas, USA*
[f]*Laboratoire GBCM, EA 7528, CNAM, Hésam Université, Paris, France*
[g]*Sorbonne Université, LJLL, UMR 7598 CNRS, Paris, France*
[h]*Institut Universitaire de France, Paris, France*

† Electronic supplementary information (ESI) available. See DOI: 10.1039/d1sc00145k

‡ These authors contributed equally to this work.

targets for drugs. Among the different classes of proteins that have been experimentally characterized, the main protease[3] is essential for processing the precursor polyprotein for the replication of the virus. Indeed, proteases are responsible for activating viral proteins for particle assembly. Due to their importance within the replication cycle of the virus, they have been proven to be successful targets for antiviral agents and are used to treat many diseases including HIV and hepatitis.[4] In the case of SARS-CoV-2, the main protease is called M^{pro} or $3CL^{pro}$. Many efforts have been made to refine the crystallographic structure of M^{pro} as the number of experimental structures available in the Protein Data Bank is increasing. While more than one hundred M^{pro} structures exist and massive efforts to discover a successful inhibitor are underway, computational approaches involving virtual screening and Molecular Dynamics (MD) simulations are needed to help experimentalists to *in silico* optimize their millions of test molecules.[5–8]

Molecular Dynamics is a powerful tool for understanding the structural and dynamical details of complex biological systems. It also enhances the ability to identify promising protein inhibitors. Two main research groups, DE Shaw Research (DESRES) and RIKEN Center for Biosystems Dynamics Research, recently released multi-microsecond MD simulations of the M^{pro} dimer.[5,6] These MD conformational ensembles both used non-polarizable force fields (n-PFFs) including DES-AMBER[9] and AMBER14ff.[10] Although the simulations are of great help for the scientific community, conventional MD (cMD) simulation results are limited by the daunting complexity of M^{pro}'s conformational space, which requires very large computational resources. In practice, both DESRES and RIKEN results were obtained on special-purpose petascale supercomputers designed for MD (Anton[11] and MD-GRAPE-4A[12] for DESRES and RIKEN, respectively). So, what can be done next? Besides these large scale MD simulations, the question of accuracy still remains open. Indeed, conformational space sampling depends by definition also on the force field used for the simulations. Our group has been involved for many years in the demonstration of the importance of considering explicit many-body effects in classical MD and free energy methods through the use of polarizable force fields (PFFs).[13–17] Indeed, electronic polarization affects solvation and modifies the stability of secondary and quaternary structures of proteins, playing therefore a crucial role in defining the conformational space of a protein. Applying such methods to COVID-19 research could provide additional insights for drug modelers and experimental teams. When our project started (end of March 2020) in response to the international High-Performance Computing (HPC) global effort to mitigate the impact of the COVID-19 pandemic,[18–20] performing long timescale MD simulations using new generations of PFFs on SARS-CoV-2 proteins encompassing hundreds of thousands of atoms (or more), such as M^{pro}, was out of reach of generalist supercomputers. Such simulations would have required years of computation.

To overcome these limitations we introduce a density-driven unsupervised adaptive sampling method based on statistical models and principal component analysis (PCA). It has been deployed on a generalist supercomputer. Since the global exploration problem is decomposed into a set of separate MD trajectories, the process can be restarted using an iterative selection method, and various computations can take place on a large number of Graphics Processing Units (GPUs) that are now available in generalist supercomputers. Such a strategy enables the Tinker-HP package,[21] which recently proposed a GPU-accelerated implementation,[22] to perform multi-microsecond MD simulations within a few days, where years would have been required with single GPU card or CPU-based conventional MD simulations. We additionally provide the capability to re-weight our simulations, which enables full exploitation of the total amount of MD trajectories to compute statistical properties that can therefore benefit from the long simulations. After describing our sampling strategy, we will detail our conformational space exploration results that notably expand over those obtained by other groups. We will unveil critical structural behavior not fully captured with n-PFFs. We particularly investigated the differences in clustering results, active site volumes, cryptic pockets, key structural activation markers linked to the oxyanion hole structuring, interactions between the C-terminal chain and the active site, and solvation patterns of some key residues. The effect of pH is also discussed.

2 Unsupervised adaptive sampling strategy for exploration: exploiting pre-exascale machines and GPUs

Adaptive sampling has been used for many years and has proven to be a powerful exploration tool to study protein folding and dynamics, ligand binding and a variety of rare molecular events.[23–26] For this family of approaches, multiple iterations of independent molecular dynamics simulations are performed, basing the initial conditions at each iteration on the results of previous iteration steps. We propose here a new unsupervised (*i.e.* fully automated) adaptive sampling strategy dedicated to our specific use of PFFs within large supercomputer systems allowing for the simultaneous use of hundreds or thousands of GPU cards. This characteristic is important as it allows us to benefit from the full potential of pre-exascale supercomputers, and will naturally transfer to future exascale machines. The results presented here benefit from a GPU acceleration in the newly developed Tinker-HP GPU code[22] that was first used here for COVID-19 simulations. However the procedure is completely general and can be applied to any homogeneous or heterogeneous computational platforms compatible with Tinker-HP[21,27] or any MD software. Therefore, in view of the particular distribution of available numerical resources, the simulations are organized by iterations as follows. At the beginning of each iteration, some initial structures are selected among the configurations sampled in the past iterations, from which independent MD simulations are run, generating new configurations. The selection of the initial structures at each iteration follows an adaptive procedure designed to enhance the exploration of a low-dimensional space of slow variables.

More precisely, M_k denotes the number of configurations available at the beginning of iteration $k \geq 0$, and $(q_i)_{1 \leq i \leq M_k}$ the configurations. Here, a configuration means the positions $q \in \mathbb{R}^{3N}$ of all the atoms of the system. In particular, at the very beginning of the algorithm, we suppose that we start with $M_0 \geq 1$ configurations, obtained from an initial conventional MD simulation (which is in practice non-polarizable), or previously available studies. At the beginning of iteration k, first, the protein is aligned in all configurations, using the backbone atoms of the 6LU7 crystal structure from the Protein Data Bank.[3] A principal component analysis (PCA)[28] is then performed, using the scikit-learn[29] and MDTraj[30] packages, on the protein atoms $(q_i)_{1 \leq i \leq M_k}$, from which the $n = 4$ principal modes are considered. This choice was made after a global analysis of the first 20 PCA modes of the first AMOEBA 0.14 μs which showed that $n > 4$ modes had variance contributions below 4% (Fig. 1, ESI†). This has also been corroborated by an analysis of RIKEN and DESRES trajectories, for which, respectively, 3 and 4 PCA modes are above 4% (Fig. 2, ESI†). We denote by $\xi_k : \mathbb{R}^{3N} \to \mathbb{R}^n$ the orthogonal projection on these n principal modes and we write $x_i = \xi_k(q_i)$. At the beginning of iteration k, this represents the current guess of slow variables of the system, and in order to enhance the sampling, we would like to explore all the values of these slow variables. In other words, ideally, we would like the values of x sampled to be uniformly distributed over some compact set of $\mathbb{R}^n$. The selection procedure is designed to push the exploration in the direction of this ideal target.

The density ρ_k of the collective variables is approximated by a Gaussian kernel, *i.e.* for $x \in \mathbb{R}^n$

$$\rho_k(x) = \frac{1}{(2\pi\sigma^2)^{n/2} M_k} \sum_{i=1}^{M_k} \exp\left(- \frac{|x - x_i|^2}{2\sigma^2} \right),$$

for some $\sigma > 0$. In practice we used the D.W. Scott method, implemented in Scipy,[31] to estimate a suitable bandwidth σ. Denoted by s_k is the number of MD trajectories that are going to be run during iteration k. In order to select the initial structures $(q_{I_1}, \ldots, q_{I_{s_k}})$ of these simulations, the indexes $I_1, \ldots, I_{s_k}$ are generated as independent random variables in $\{1, \ldots, M_k\}$ distributed according to

$$\mathbb{P}(I = i) = \frac{{\rho_k}^{-1}(x_i)}{\sum_{j=1}^{M_k} {\rho_k}^{-1}(x_j)}.$$

In other words, among all the structures currently available, q_i is selected to be the initial structure of a new simulation with a probability inversely proportional to its density (in the low-dimensional space given by the first four PCA components). The effect of this selection can intuitively be illustrated as follows: if two domains of similar size (in the sense of the Lebesgue measure on $\mathbb{R}^n$) have been visited, with one that concentrates most of the past trajectories while the other contains only a few points, then approximately half of the new initial structures will be selected in each domain; in contrast, a uniform selection among the past configurations would have put much more weight on the dense domain.

From the initial structures $(q_{I_1}, \ldots, q_{I_{s_k}})$, s_k independent MD simulations are sampled, and the state of each simulation is recorded every 0.1 ns (the initial structure is not recorded, since it has already been recorded in one of the past iterations). Here, independent means that the initial velocities (sampled according to the equilibrium Gaussian density) and the white noises of the Langevin thermostats are independent (and, of course, independent from previous iterations, so that a trajectory starting at some configuration q_i will be different from the trajectory that initially produced this q_i). At the end of this kth iteration, structures $(q_j)_{M_k < j \leq M_{k+1}}$ have been added, and iteration $k + 1$ starts.

The procedure penalizes areas that have already been extensively visited, and is in a way reminiscent of the metadynamics[32] method except that the statistical biasing is done through a selection step between each iteration rather than a biasing force updated along the trajectory. By comparison with metadynamics, this unsupervised selection step has the advantage of overcoming the critical choice of initial collective variable at the beginning of the simulation reinforcing automation of the sampling scheme.

This strategy belongs to the family of counts based adaptive sampling algorithms, where one only exploits the number of passages in the different states (micro or macro) visited in the previous iterations to choose which state to restart trajectories from. These are known to be efficient for pure exploration purposes (as is the case here), even though more refined algorithms exist when some information is available as to where the sampling should be guided.[24] However, in contrast to what is usually done in the context of Markov State Models (MSMs),[23] the states are not defined by applying a clustering algorithm to the already explored structures, but are the projection on the n principal components generated by PCA (here, $n = 4$ as we discussed) of all the previous data. This has the advantage of providing an unsupervised sampling strategy that does not rely on a particular clustering algorithm (and therefore its associated parameters) and treating every point of this 4-dimensional representation differently.

At the end of the simulation, M_K configurations have been sampled with K, the total number of iterations. For a large K, the distribution of these configurations does not converge to the canonical distribution because of the statistical bias induced by the selection. To compute thermodynamic quantities, this bias should be taken into account. In that case, we interpret the previous selection as an importance sampling scheme. Thus, we have to compute a score $\omega_i > 0$ for each $i \in \{1, \ldots, M_K\}$ so that the canonical average of an observable φ is estimated by

$$\langle \varphi \rangle \simeq \frac{\sum_{i=1}^{M_K} \omega_i \varphi(q_i)}{\sum_{i=1}^{M_K} \omega_i}.$$

The score ω_i is the ratio between the probabilities to obtain q_i in the biased simulation and in an unbiased simulation (where, between each iteration, the next initial conditions are uniformly

chosen among all currently available configurations, *i.e.* all with probability $1/M_k$). As a consequence, it is computed as follows: for all $i \leq M_0$, $\omega_i = 1$. Suppose by induction that ω_i has been computed for all $i \leq M_{k-1}$ for some k. Let $(i_1, \ldots, i_{s_k})$ be the indexes that have been randomly selected for the initial conditions at the beginning of iteration k. For each $h \in \{i_1, \ldots, i_{s_k}\}$, α_h is computed:

$$\alpha_h = \frac{1}{M_k \mathbb{P}(I = h)} = \frac{\rho_k(x_h)}{M_k} \sum_{j=1}^{M_k} \rho_k{}^{-1}(x_j).$$

Then, the score of all the configurations that are generated during iteration k from the initial condition q_h is $\alpha_h \omega_h$. That way, ω_i is computed for all $i \leq M_k$.

This latest point is important since it means that the total simulation time can be used to compute average statistical properties that are unbiased and therefore exploitable. For example, it is possible to compare them to those obtained upon performing conventional MD runs.

Finally, it should be noticed that, instead of the PCA, this adaptive sampling strategy may be used with any other collective variables and/or dimensionality reduction algorithm. Overall the procedure is fully unsupervised, fast and can be used within Tinker-HP in a fully automated way.

3 Large scale unsupervised adaptive simulation using polarizable force fields (PFFs) and GPUs

3.1 Preparation of systems and choice of initial structures

In order to perform a large scale unsupervised adaptive sampling simulation, starting structures have to be selected from a conventional MD simulation (using either n-PFF or PFF approaches). We chose the RIKEN dataset as the starting point. From their 10 μs conventional MD simulation (PDB: 6LU7, pH = 8)[3] using the n-PFF AMBER14ff[10] approach and using PCA as a guiding thread, we carefully extracted 14 relevant structures that represent our starting point for the study. It is worth noting that the 6LU7 crystal structure is a holo structure including a covalently bound inhibitor. The inhibitor-unbound apo structure was initially obtained by RIKEN removing the inhibitor and relaxed over 10 μs of simulation (https://data.mendeley.com/datasets/vpps4vhryg/1). Each Amber14ff structure was then minimized with the AMOEBA PFF[33–36] and an L-BFGS algorithm until a Root Mean Square (RMS) of 1 kcal mol^{-1} on the gradient was reached. It is important to note that not all histidine residues are protonated in the RIKEN structure similarly to the DESRES one. Since it has been recently demonstrated that the highest pK_a for possible protonation of histidine sites was lower in the SARS-CoV-2 M^{pro} than in the SARS-CoV-1 M^{pro}, being about 6.6,[37] the present simulation is therefore consistent with physiological pH conditions (pH = 7.4).[38]

3.2 Simulation protocol

The presented all-atom simulation was performed using the newly developed GPU module[22] within the Tinker-HP package,[21] which is part of the Tinker 8 platform.[39] This newly developed module is able to efficiently exploit mixed precision[22] offering a strong acceleration of simulations using GPUs. The 98 694 atom initial structure of the fully solvated M^{pro} dimer was extracted from the Protein Data Bank (PDB: 6LU7) and the AMOEBA PFF[33,34,36] was used to describe all atoms (protein and water). Periodic boundary conditions using a cubic box with side lengths of 100 Å were used. Langevin molecular dynamics simulations were performed using the BAOAB-RESPA1 integrator[40] using a 10 fs outer timestep, a preconditioned conjugate gradient polarization solver (with a 10^{-5} convergence threshold), hydrogen-mass repartitioning (HMR) and random initial velocities. Periodic boundary conditions (PBCs) were employed using the Smooth Particle Mesh Ewald (SPME) method with a grid of dimensions 128 Å × 128 Å × 128 Å. The Ewald-cutoff was taken to be 7 Å and the van der Waals cutoff to be 9 Å. As we explained, we started the simulation by running a 10 ns cMD for each of RIKEN's 14 representative structures (as mentioned in Section 3.1). A first adaptive sampling selection was then conducted on those 140 ns initial structures. We chose to use the first four PCA components (see the method section) as conformational space for the adaptive sampling method. At each iteration, the adaptive sampling procedure is then used on these newly computed first four PCA components in order to select 100 structures. Then, 100 independent molecular simulations of 10 ns were performed in the NVT ensemble at 300 K on single NVIDIA V100 GPU cards. Each trajectory belonging to the same adaptive sampling iteration was run simultaneously on the HPE Jean Zay Supercomputer (IDRIS, GENCI, France). A single adaptive sampling iteration took less than 18 hours to complete, allowing a production rate of 15.14 μs in two weeks. Overall, the simulations ran over 12 working days in line with computer center resources availability.

The complete 15.14 μs trajectories with and without water are freely accessible through the Swiss National Supercomputing Center (CSCS)[41] and have been linked to the BioExcel/Molssi COVID-19 community portal. A movie depicting the progress of the exploration can be found in the ESI.†

3.3 Performance of the adaptive sampling exploration: comparisons with other available simulations

As we mentioned in the method section, we use the PCA[28] as an intermediate quantity to orient the consecutive sampling iteration. However, it is also a good quantity to quickly assess the performance of the adaptive sampling scheme for the exploration of the conformational space. Indeed, the analysis of MD trajectories with PCA is a well-known strategy known in the community as the "essential dynamics".[42–44] PCA, being a dimensionality reduction algorithm that evaluates directions maximizing the variance of the dataset, is thus a revealer of a system conformational diversity. Therefore, it can be seen as a way to assess the amount of sampling and can also detect explicit "essential motions" otherwise not discernible using predefined collective variables. Thus, it is interesting to compare the amount of sampling on the space of these reduced variables. This is why we projected the RIKEN, the DESRES and

the first 2 μs Tinker-HP data set on the first two PCA components of the first 2 μs of the Tinker-HP data set (Fig. 1a and b). One can see that, in this space, the Tinker-HP adaptive scheme already captured the RIKEN and DESRES major main PCA features. It also appears that the RIKEN trajectory sampled a portion of conformational space close to the Tinker-HP data set while the DESRES trajectory seems to explore only the area that is most sampled by Tinker-HP. The same procedure was applied for the PCA components and associated data of the entire Tinker-HP data set (Fig. 1c and d) and it is striking that a much larger portion of conformational space has been sampled by our adaptive scheme. Additionally, we also projected the same data sets on the first two principal components of the RIKEN trajectory which gives the same justification of the larger sampling obtained by our method (see Fig. 4 in the ESI†).

As a preliminary conclusion, we can say that our adaptive sampling strategy allowed us to generate a multi-microsecond polarizable MD simulation that sampled a vast area of the free energy landscape. In addition, we analyzed the Root Mean Square Deviation (RMSD) on protein backbones *versus* the radius of gyration (see Fig. 5 in the ESI†) for the AMOEBA 15.14 μs. It revealed large conformational changes. Variations for the radius of gyration are about 2 Å, while the variation is 1 Å for non-polarizable conventional MD. Such plots are very useful to understand one key question: what makes the AMOEBA results different? Is it the choice of PFF (*vs.* n-PFF) or is it the choice of adaptive sampling strategy. In order to provide a fair (and somewhat quantitative) comparison between the FFs and to decorrelate the effects of the FFs themselves from the gains due to adaptive sampling, we limit ourselves to structures with a reweighting score (see the section above) greater than 1 as it is the score of the frames visited during a conventional MD simulation and as frames with scores lower than 1 are the ones that have been favored by the adaptive algorithm to maximize exploration. 3/4 of the points are therefore removed using this criterion offering a view of the performance of the adaptive sampling. The plot representing the remaining point is presented in Fig. 3 (ESI)† for AMOEBA and it can be directly compared to the RIKEN plot for example. Clearly differences exist between AMBER and AMOEBA results, and they also come from the choice of FF. In addition, important changes are also observed in different important areas of the protease such as the dimerization site. The RMSD of the protein backbone *versus* the RMSD of the chain A dimerization site (see Fig. 6 in the ESI†) depicts large fluctuations between 6 and 7 Å. DESRES and RIKEN trajectories exhibited only 2 Å, which is in the order of the size of the observed PCA features. Overall, these first observations of the differences between the non-polarizable and the polarizable simulations motivate a further analysis of the different simulations.

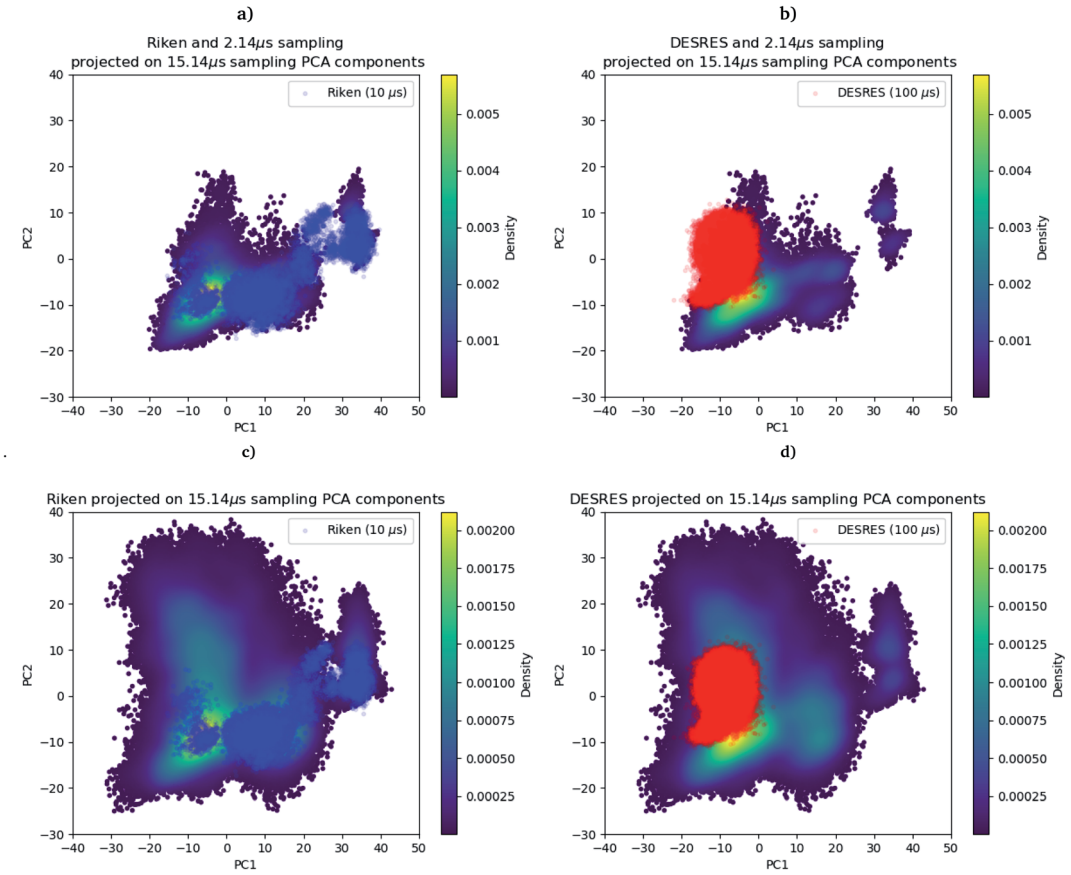


Fig. 1 RIKEN and DESRES datasets superposed on the 6LU7 protein backbone and projected on the first two PCA components fitted to, respectively, the 2 μs (a and b) and 15.14 μs (c and d) simulations.

3.4 Unsupervised clustering and extraction of the unbiased relative free energy between representative domains

First, if the PCA analysis reveals useful information, a proper clustering of the produced ensembles is a more precise and quantitative framework to discuss differences between simulations and possible new features captured by the AMOEBA force field. Therefore, we applied to all trajectories the density-based spatial clustering of applications with the noise (DBSCAN) method.[45] DBSCAN is an unsupervised machine learning algorithm that groups together data in clusters according to their density. It has the particularity to label points as noise if they are not in a dense region and are then not assigned to any cluster. DBSCAN is particularly well suited in our case as it is especially designed to target arbitrary shape clusters. To evaluate the density, DBSCAN uses two parameters, ε the distance at which two points are considered to be neighbors and MinPts the minimum number of points needed to define a cluster. ε was chosen using the nearest neighbor graph procedure, *i.e.* by plotting the distance to the nearest n-neighbor for each point, ordered from the largest to the smallest value, and evaluating ε for which the graph starts forming an elbow. For a given ε we then scanned different values of MinPts until relatively large clusters covering a wide range of the space are found. In practice we evaluated the distance to the 4th nearest neighbor on the 4 dimensions composed of the first four 15.14 μs principal components generated by PCA (see Fig. 7 in the ESI†). For DESRES and RIKEN, after being aligned to their respective PDB, the structures were projected on this 4D space.

Our choice of using the AMOEBA 15.14 μs PCA components as the starting point of the clustering is driven by the conformational diversity brought about by the coupling of the PFF and the adaptive sampling scheme. For visualization, clusters are then projected on the first two principal components (Fig. 2). To evaluate the quality of the clustering we used three scoring methods for unknown labeled data:[46] Silhouette coefficient, Calinski–Harabasz and Davies–Bouldin indices. These indices confirmed our parameter optimization procedure and the high quality of the clustering. Our new adaptive sampling scheme has the main advantage of offering access to true statistical properties such as free energies. To understand the cluster stability, the free energies for each cluster are computed (Fig. 3c and d) through the evaluation of the probability distribution over the total number of structures. Notice that, since not all the structures are part of a cluster, the cluster probabilities do not add up to one. The unbiased probability distribution (Fig. 3a and b) is estimated with the de-biasing procedure explained in the previous section. The de-biasing step preserves the trend

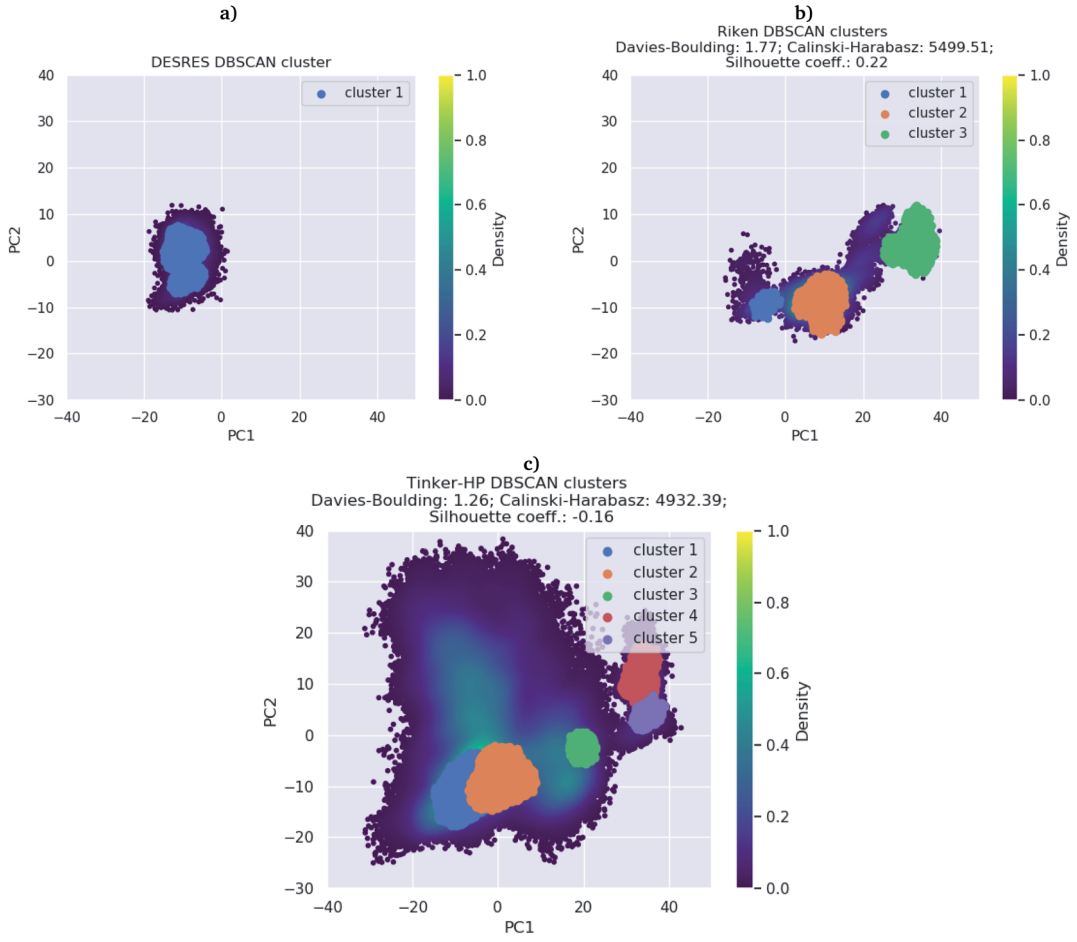


Fig. 2 DBSCAN clustering of (a) DESRES (100 μs) and (b) RIKEN (10 μs) datasets and (c) the Tinker-HP 15 μs simulation.

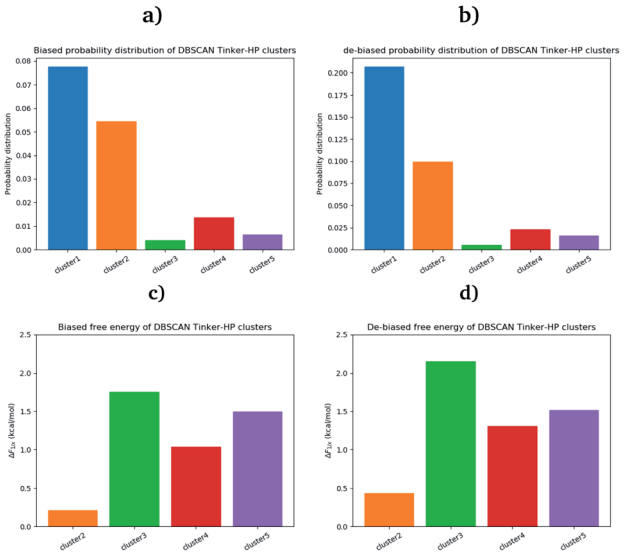


Fig. 3 Biased (a) and unbiased (b) probability distribution of DBSCAN Tinker-HP clusters. Biased (c) and unbiased (d) relative free energies of the DBSCAN Tinker-HP 15.14 μs clusters, with respect to cluster 1.

between clusters but increases the probabilities. It means that the five clusters were disadvantaged by the adaptive sampling. For example, the biased simulation assessed an 8% probability for the presence of cluster 1, which should have contained, in an unbiased simulation, 20% of the configurations. Besides, cluster 1 is indeed the most explored region by both DESRES and RIKEN. Hence, the algorithm managed to disadvantage this part of the conformational space which is what we could have expected as it favored intermediate transition areas to the detriment of dense regions in order to discover new regions. The effect of the polarizability on structural properties such as volumes and RMSF is further depicted in the next section. Overall, our approach demonstrated our capability to reach high-resolution conformational space exploration using a PFF. We identified 5 different clusters using AMOEBA (see Fig. 2). While some of these states were already identified in previous n-PFF simulations (RIKEN and DESRES), we found two new non-negligible conformations (according to Fig. 3) that can be critical, *e.g.*, for the computation of thermodynamic properties and finally guide further ensemble docking simulations and/or to help to interpret experimental results.

4 Correlation with experimental data: structural markers for protomer activity and new features

4.1 Markers of the structuring of the oxyanion hole

To ensure the validity of our AMOEBA simulations, we compared our computed properties with available experimental data. Since the beginning of the COVID-19 pandemic various X-ray structures have been released (PDB: 6Y84, 6LU7, 6Y2G, …).[3,47,48] They provided important insight on specific interactions between residues as well as structural information about the active site. To be consistent with RIKEN simulations we used as reference the same PDB: 6LU7.[3] Note that DESRES used another PDB, 6Y84,[47] which we used as a reference in the computation of its properties. Crystal structures have been projected on the first two PCA components of the Tinker-HP simulations (see Fig. 8 in the ESI†).

Recently, Zhou *et al.* published an experimental study of the apo structure (PDB 1UJ1)[49] at physiological pH. They found several features allowing for the characterization of the presence of the oxyanion hole structure which is a key structural element of the activity of each protomer. In particular, they proposed to monitor the distance between Glu166 and His172 and the π–π stacking between Phe140 and His163. The definitions of these structural markers are not new and were initially also discussed for the SARS-CoV-1 M^{pro}.[50,51] The oxyanion hole is responsible for the stabilization of the substrate in the active site and is of crucial importance for the enzyme's kinetics and activity. Indeed, the substrate binding site is composed of 4 pockets labelled S1 to S4 with the S1 pocket involving very conserved residues such as Glu166, His172, His163 and Phe140. The oxyanion hole of the cysteine protease encompasses backbone amides (Gly143, Ser144, and Cys145) while residues 138 to 145 form the so-called oxyanion-binding loop.[48,51,52] The existence of this latter is responsible in part for the structuring of the S1 pocket.[51] When the stacking and the Glu166–His172 interaction are broken, a rearrangement occurs leading eventually to the collapse of the oxyanion hole. In this case, Glu166 potentially interacts with His163 instead of His172. In other words, strong interactions of Glu166 with His172 associated with a Phe140–His163 stacking are consistent with a structured oxyanion hole, and can be used as a marker of the activation of the enzyme protomer. Inversely, a strong interaction of Glu166 with His163 would rather be a marker of the protomer inactivation linked with a collapse of the S1 substrate-binding pocket. Of course, such analysis is only interpretative, the oxyanion hole structuring being far more complex. However, it has been shown to be useful since the initial studies on the SARS-CoV-1 main protease.[51] In practice, the absence of a well-structured oxyanion hole leads to the inhibition of the enzyme's activity. Experimentally, it is known that the M^{pro} monomeric form is inactive while the active form is a homodimer containing two protomers.[53] In the holo state of SARS-CoV-1, the first protomer is active while the second one is found inactive.[54] For SARS-CoV-2, a pH = 6 crystal structure (PDB: 1UJ1)[53] predicted a strong asymmetry of the protomers with an inactive conformation for one of the protomers linked to a broken Glu166 and His172 interaction. However, the inactivity of one of the protomers is still a hypothesis as crystallographic studies of the dimer in the space group *C*2 encounter difficulties in capturing the details of each individual protomer. Indeed, data are only available on one of the protomers in the asymmetric unit which always leads to the more ordered conformation and therefore to the most active one. Concerning the apo state, recent experimental results lead to a potential low activity of the apo dimer linked with an observed destructured oxyanion hole.[49] It is important to point out that distances/markers exhibit a distribution of

different values centered around a maximum of frequency due to the liquid conditions that differ from the crystal ones (Fig. 4).

Then we investigated these markers. To study the Phe140–His163 stacking interaction, we use a stacking-index developed by Branduardi and Parrinello[55] who described it as a product of 2 Fermi functions, one considering the radial dependence, and the other the angular dependence of the interaction. The model provides an index ranging from 0 for a non-stacked interaction to 0.6 for a perfect one. The Glu166 interactions and π–π stacking were thus calculated for both chains of all RIKEN, DESRES and Tinker-HP structures and then classified into histograms. Finally, each histogram has been unbiased (*i.e.* reweighted) and extrapolated using a univariate kernel density estimator. Final results are given in Fig. 9 of the ESI.† Furthermore a 6 μs adaptive sampling simulation was performed (on the Irene Joliot Curie Machine (TGCC, GENCI, France)) on the monomer species (PDB: 6LU7) and the same features as discussed below (π–π stacking between Phe140 and His163, and Glu166 interactions with both His172 and His163) were calculated. Since the monomer is known to be in an inactive conformation, it helps us to rationalize the behavior observed in our simulations. Results are depicted in Fig. 10 in the ESI.† The preparation and simulation protocols are similar to what we did for the dimer. Therefore, since His172 and His163 are also unprotonated, we minimized the structure up to a RMS of the gradient of 1 kcal mol^{-1} and generated an initial cMD of 200 ns. We then selected 100 random initial structures according to the Adaptive Sampling protocol of structure selection using the PCA, and we performed 6 iterations of 1 μs for a total simulation time of 6 μs.

For the interaction formed by Glu166, in the case of Tinker-HP, we observed an asymmetry between the two protomers. In one protomer the Glu166–His172 interaction is significantly weaker than in the other exhibiting a well-defined marker of a smaller activity of the protomer. This relative non-interaction is in accordance with the results obtained on the monomer which appears to be similar (see ESI Fig. 10†). The situation is more complex in the other protomer where we observe an oscillation between two states, presenting either a formed Glu166–His172 interaction or its absence leading to only some partial activity markers. However, the "interacting" state clearly dominates the statistics. These results demonstrate that the oxyanion hole is only partially organized in the other protomer. This is consistent with experimental data on the apo state[49] and also with the data on the active protomer of the holo state which shows distances of around 5 Å (see ref. 37 and references therein for a discussion of the different available crystal structures). It is, of course, only one single marker but it could already corroborate the asymmetry observed in the holo state where only one protomer is found to be active,[48] a similar feature to what was previously observed in SARS-CoV-1.[54] Based on the analysis of this single marker, we tend to have an inactive first protomer coupled to a second protomer that exhibits some partial but clear activity features (two states) when compared to its inactive counterpart and to the monomer. Similar interpretations can be deduced from the DESRES and RIKEN simulations despite a less clear picture of the His172–Glu166 interactions which appear extremely flexible with more mixed states, especially for AMBER. This is not surprising as Glu–His interactions can be classified as H-bonds, a class of directional weak interactions that are known to be difficult to model using n-PFFs[56,57] as polarizability contributes significantly to the accuracy of simulations of structures with hydrogen bonds.[15,58] However, a single distance is not enough to reach a conclusion and should be combined with other markers such as the Glu166–His163 distance. We note here a stronger asymmetry of such distances in protomers for DESRES while in the case of RIKEN and Tinker-HP we could again observe a mixture between interacting/non-interacting states. However, this second marker should be carefully considered as a direct comparison with our monomer simulation (see ESI Fig. 10†) shows that this distance criterion is less well-defined for discussing the protomer "activity" than the Glu166–His172 distance. Since our monomer is known to be inactive, it could be deduced that this marker should always be associated with the evaluation of the Glu166–His172 distance. In practice, one should look at the relative strength of these interactions and the Glu166–His163 distance here appears to be clearly longer than the Glu166–His172 ones. Glu166–His163 distances appear consistent with data on the active protomer of the holo state which shows distances going beyond 6–8 Å (see ref. 37 and references therein for a discussion of the different available crystal structures). In that connection, a better conservation of the catalytic dial is observed in the RIKEN and Tinker-HP simulations with a smaller Cys145–His41 distance compared to DESRES (see ESI Fig. 9†). The active site of the M^{pro} protease comprises a catalytic dyad composed of residues Cys145 and His41. X-ray crystal structures of SARS-CoV-1 (ref. 51 and 52) found a Cys145–His41 distance between 3 and 3.9 Å. In comparison, our simulations revealed distances of around 4 Å while AMBER and DES-AMBER distances are, respectively, around 4.5 and 6–7 Å. Regarding the relatively small differences between the SARS-CoV-1 and SARS-CoV-2 main proteases, AMOEBA results appear closer to experimental data.

Finally, a last marker is studied to confirm our observations: the π–π stacking between Phe140 and His163. Results are

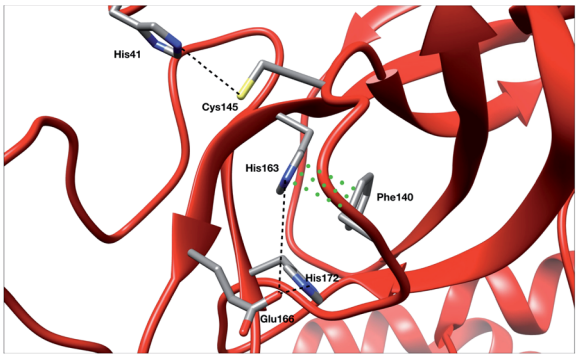


Fig. 4 Representation of the π–π stacking interaction between His163 and Phe140 residues (green points) and of several distances of interest which are responsible for the stability of the active site (black dashed lines).

depicted in Fig. 9 in the ESI.† Tinker-HP does not capture this stacking in one protomer while again two mixed-states (stacked and un-stacked) are observed in the other protomer. The same observations can be made for DESRES and RIKEN although the states are less well defined in connection with the well-known difficulty of capturing π–π stacking with n-PFFs.[59] Despite these differences, the 3 simulations appear consistent. Overall, our initial conclusion stands: we describe an asymmetric situation where one protomer is fully inactive and the other shows some partial activity features. It is important to point out that these results are not artificial and linked to our starting structure. Fig. 11 of the ESI† shows the convergence of the stacking

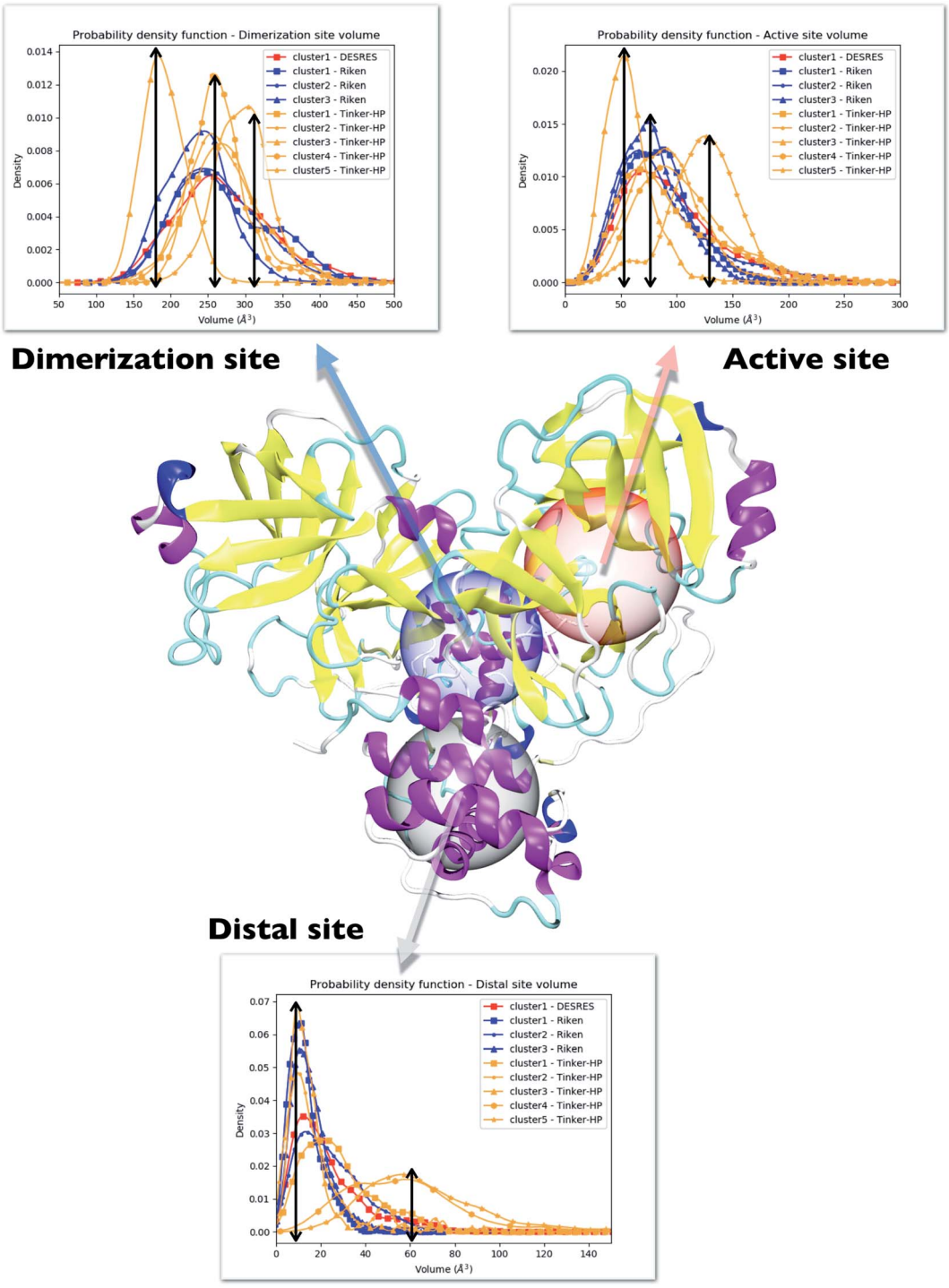


Fig. 5 Representation of the 3 cavities considered in this study: the dimerization site, active site and distal site. For each cavity, trends inferred from each cluster are depicted and superposed on three different graphs. Each curve has been unbiased according to the reweighting approach described in this work. Cavity volumes are the sum of volumes found in both protomers. The black arrows link the maxima of frequency to the volume axis to highlight the difference between clusters.

marker over the 15.14 μs simulation. If protomer 1 is clearly not evolving over the simulation, protomer 2 evolves slowly towards the discussed 2 state organization. Overall, our results are compatible with the description of the apo crystal structure by Zhou *et al.*[49] who observed an incomplete structured oxyanion hole exhibiting several mixed states of structuring. This highlights the large flexibility of the enzyme discussed in the experimental literature at room temperature.[38] Our data also support the possible strong asymmetry between protomers discussed in the holo state.[53]

4.2 Evaluation of the volumes of the enzyme cavities

One way to measure some potential global differences between the different simulations is to measure the active site volume in each cluster and to depict the observed trend similarly to the π–π stacking previously. Besides the main active site cavity, the main protease exhibits 2 other cavities: the distal site and the dimerization site. Represented in Fig. 5, these cavities are considered as potential targets for drug inhibition.[60,61] An accurate description of each of these cavities is essential to the estimation of efficient inhibitors. For each cluster of each dataset, we thus estimated those 3 cavity volumes. Volumes were calculated for each isolated cluster using POVME 3.0 software.[62] For each cavity, a 1.0 Å grid spacing was chosen. Residues 7–198 and 198–306 and all residues within 3.5 Å from the other protomer were selected for the active, distal and dimerization sites with, respectively, 12 Å, 10 Å and 10 Å. 1000 structures were randomly chosen per cluster for the analysis. When a cluster had less than 1000 structures, we chose all the structures. Detailed information is given in the ESI† on the size of each cluster as well as their relative size (see Table 1 in the ESI†). Similarly to the π–π stacking and the Glu166 distances, we used the univariate kernel density estimator on the volumes. The final volumes are depicted in Fig. 5. Additionally, each cluster has a normal distribution supporting the quality of DBSCAN clusters. Different trends appear, represented by black arrows. For the 3 cavities, we observed a similarity between the single DESRES cluster, clusters 1 and 2 from RIKEN and Tinker-HP's clusters 1 and 2. Agreement is also found with volumes obtained by Sztain *et al.* using a Gaussian accelerated MD (GaMD) enhanced sampling strategy coupled with AMBER ff14SB[8] which also match these results confirming the importance of simulating long enough in conventional MD. Overall, while Tinker-HP clusters 1 and 2 are in good agreement with RIKEN and DESRES clusters, our clusters 3, 4 and 5 appear to be different and specifically highlight the importance of the PFF choice, *i.e.* these data are not obtained using enhanced sampling coupled with non-PFFs.[8] As we pointed out earlier, differences indeed occur between clusters and between different datasets, going in the same direction of the previous analysis of the π–π stacking between residues Phe140 and His163 in chains A and B. For Tinker-HP, we observed a contraction for the three cavities in cluster 3 while in cluster 4 and especially cluster 5, we observed a strong difference with a non-negligible increase of the cavity volumes. Cavities from clusters 4/5 depict stronger volume fluctuations when using the AMOEBA PFF. While cavity volumes obtained from AMBER/DES-AMBER simulations and from clusters 1 and 2 from AMOEBA simulations are in agreement, the AMOEBA results clearly capture an additional feature not captured by the DES-AMBER and AMBER simulations. This information could be important for designing potential new inhibitors.

Consequently, since strong differences between methods are observed in the volume evaluations of the different clusters, it is interesting to estimate the global protomer volumes if one wants to try to capture further the discussed asymmetry. Protomer volumes can be found in Fig. 6. Protomer 1 (predicted to be non-active) depicts a strong gaussian behavior while protomer 2 (predicted to be oscillating between an active and a non-active state) is characterized by a spread gaussian with more important associated volume compared to protomer 1. This increase of volume is therefore concomitant with the previous asymmetry related to the various discussed structural markers. It is worth noting that this asymmetry is also found for the DESRES simulation but to a lesser extent compared to that for the AMOEBA Tinker-HP simulations. Concerning the RIKEN dataset, this feature is not found as both protomers depict a similar gaussian trend with very similar values.

4.3 Analysis of the local fluctuations: high flexibility of the C-terminal region

Finally, it is also possible to study local fluctuations in the structural dynamics of the M^{pro} dimer system to uncover other types of difference between datasets. We calculated the fluctuation of residues in each cluster on the same 1000 previously randomly chosen structures per cluster using the Root Mean Square Fluctuation (RMSF). These were calculated on the 5 clusters from Tinker-HP (AMOEBA), the 3 clusters from RIKEN (AMBER) and the single cluster from DESRES (DES-AMBER). Results are depicted in Fig. 7. The most interesting fluctuation as well as the main differences between clusters originates from a different spatial rearrangement of the C-terminal region

Table 1 Average and standard deviation of the number of water molecules around His163 and His41 residues in DES-AMBER, AMBER and AMOEBA force field simulations (pH 7.4)

	His163		His41	
	Protomer 1	Protomer 2	Protomer 1	Protomer 2
DES-AMBER	0.14, $\sigma = 0.48$	0.77, $\sigma = 0.44$	4.01, $\sigma = 1.17$	1.61, $\sigma = 0.75$
AMBER	0.49, $\sigma = 0.57$	0.44, $\sigma = 0.41$	2.38, $\sigma = 1.11$	2.25, $\sigma = 1.23$
AMOEBA	0.31, $\sigma = 0.51$	0.13, $\sigma = 0.34$	1.48, $\sigma = 0.99$	1.62, $\sigma = 1.06$
Experiments[38,49,50,53]	0 or 1		1	

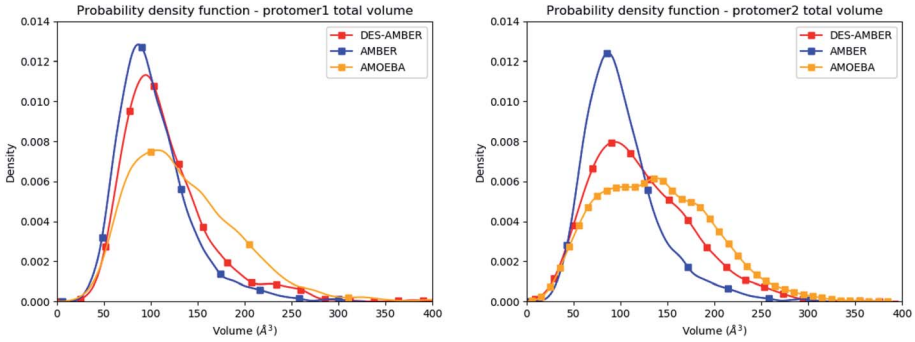


Fig. 6 Graphical representation of the distal + active sites for protomer 1 (on the left) and protomer 2 (on the right) for the DESRES, RIKEN and Tinker-HP simulations.

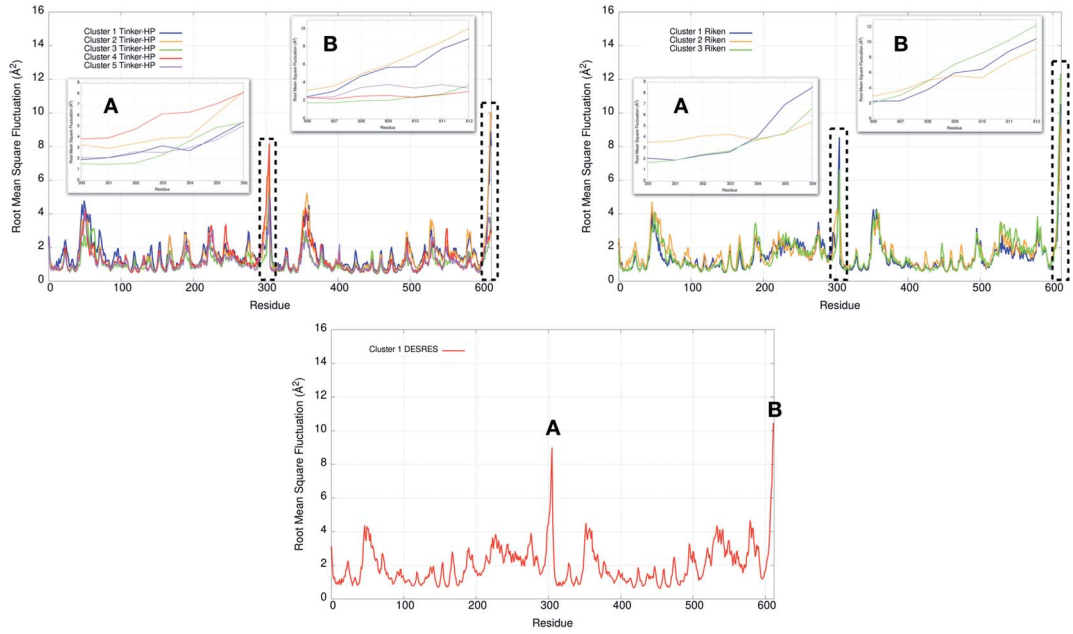


Fig. 7 Representation of the RMSF for each cluster of each simulation (Tinker-HP, RIKEN and DESRES). Zoomed-in images of both chains (A and B) are represented in subgraphics and correspond to the C-terminal end where the most important fluctuations are found (residues 300 to 306 for chains A and B).

of the protein (*e.g.* residues 300 to 306 on chains A and B of the dimer). In fact, this region is highly dynamical, which is in accordance with experimental X-ray observations where the electron density of the C-terminal domain was insufficient for backbone tracing, suggesting the flexibility of this region.[49] Visual enlargements of this region are provided in the subgraphics of Fig. 7 for chains A and B that do not differ significantly. Cluster 1 from the DESRES simulation depicts the same fluctuation as cluster 1 from the RIKEN simulation. This behaviour of the C-terminal region in these two clusters is characterized by a π–π interaction between Phe305 and His41, eventually blocking the access of any ligand to the active site. When the C terminal region does not interact with His41, it adopts an unfolded configuration which shows the high flexibility of these terminal amino acids. Structural representations can be found in Fig. 8. As this event is observed on the active site of only one chain and not both of them, it could be another marker of the previously mentioned protomer inactivation. We also observed such fluctuations in clusters 1 and 2 extracted from our Tinker-HP/AMOEBA simulations. However, in cluster 1, while the Phe305–His41 π–π interaction is indeed observed, we measure a lower fluctuation of chain A for cluster 1. It corresponds to a weaker interaction between Phe305 and His41 as configurations where the C-terminal branch is less structured are preferred. A similar feature is observed for cluster 2 of RIKEN, but with an inversion of fluctuation peaks between A and B. Overall, clusters 1 and 2 obtained from the Tinker-HP and RIKEN simulations appear relatively similar in the PCA space. They correspond to clusters where the C terminal region can oscillate between two states: one with a π–π stacking interaction between Phe305 and His41, and another with a less structured C-terminal branch with higher flexibility. Clusters 4 and 5 from our Tinker-HP simulations and to a lesser extent RIKEN's cluster 3 correspond to another configuration of the C-

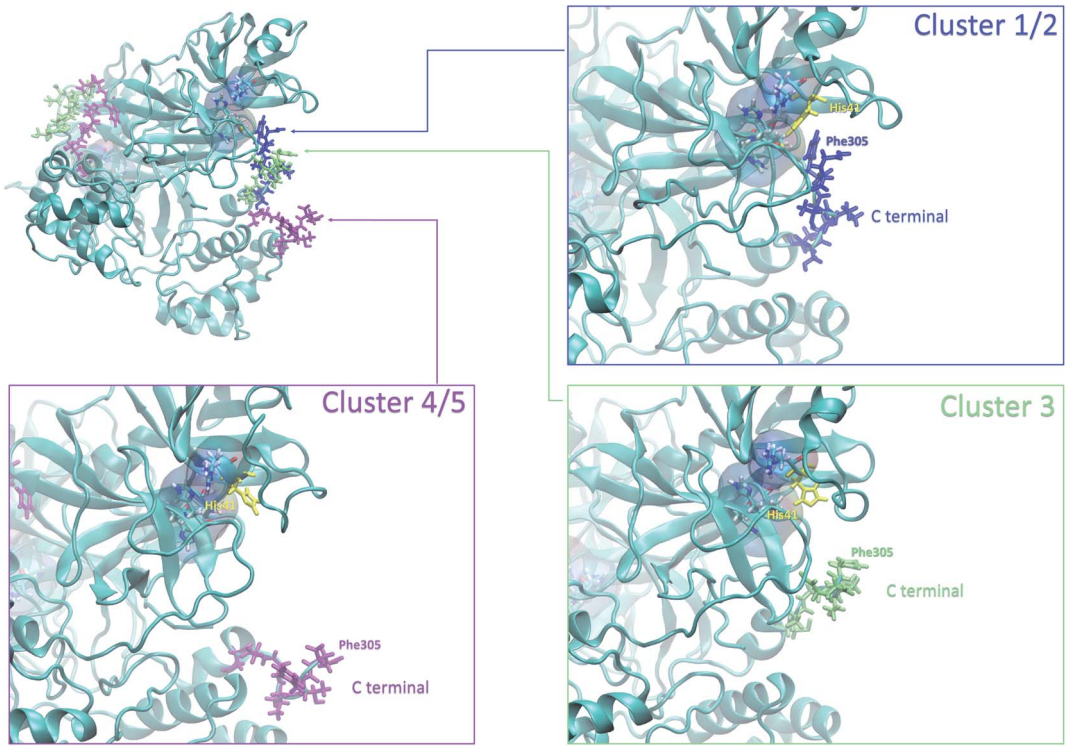


Fig. 8 Representation of the 3 possible states of the C terminal end. The whole protein is presented in ice blue. The C-terminal end presented in sapphire blue depicts most of the states in clusters 1 and 2, where the Phe305 residue of the C-terminal region is stacked with His41 of the catalytic site. The C-terminal end presented in lime depicts most of the states in cluster 3, and the one presented in purple depicts most of the states in clusters 4 and 5.

terminal region. Representative pictures are provided in Fig. 8 for each cluster C-terminal conformations. In these clusters, the C-terminal region appears more preserved/organized as it is localized further from the active site. To summarize the discussion concerning this specific feature, the high C-terminal flexibility observed in the X-ray experiments can be traced back to a modulated access to the active site linked to the absence of π–π stacking between Phe305 and His41. In other words, the C-terminal region of the fully inactive protomer is shown to oscillate between several states and one of them directly interacts with the other protomer active site. Such interaction tends to block the active site access, therefore modulating down the activity of the potentially most active site. This high flexibility is captured by both RIKEN and Tinker-HP, exemplifying the importance of the local conformational sampling and supporting the experimental analysis of a full inactivation of the apo state.[49]

5 Comparative ligandability analysis: searching for cryptic pockets

In order to check if all the previous features could affect the ligandability of the M^{pro} dimer system, we decided to search if new cryptic pockets are detected in each cluster. By taking into account the same sets as for the cavity volume analysis, cryptic pockets were searched using DoGSite Scorer software,[63] an automated tool for pocket detection and pocket descriptor calculation. DoGSite Scorer detected 18 pockets located on chain A or at the interface of chains A and B of the SARS-CoV-2 protease 6LU7 crystal structure. Among these pockets, 6 are already described in the literature:[8,64] pockets 'P_1_1', 'P_3' and 'P_15' corresponding to the dimerization site; the 'P_2' pocket corresponding to the active site and the 'P_6' and 'P_11' pockets located in the distal region. These 18 pockets were used as a reference and all pockets detected on the DESRES, RIKEN and Tinker-HP selected structures were assigned to these reference pockets by comparing the list of residues of the different pockets and selecting the reference pocket with the maximum number of common residues. When the maximum number of common residues was lower than 5, and the ratio between the maximum number of common residues and the number of residues in the predicted pocket was below 0.25, the pocket was not assigned to any reference pocket and was defined as a new cryptic pocket. New cryptic pockets were named after the first structure in which they were detected and added to the set of reference pockets. For example, the 'R_c1_s1_P14' mentioned in Fig. 11 is the pocket P_14 detected by DoGSite Scorer in structure 1 (s1) of cluster 1 (c1) of the RIKEN (R) simulations. The results of pocket assignation and new cryptic pocket identification are presented in Fig. 10. We observed that the reference pockets previously highlighted as 'active site', 'dimerization site' and 'distal site', except 'P_6', are particularly conserved and detected in a large majority of analyzed structures. However, a consequent number of other pockets were also detected: (1) in a few structures such as 'R_c1_s2_P21',

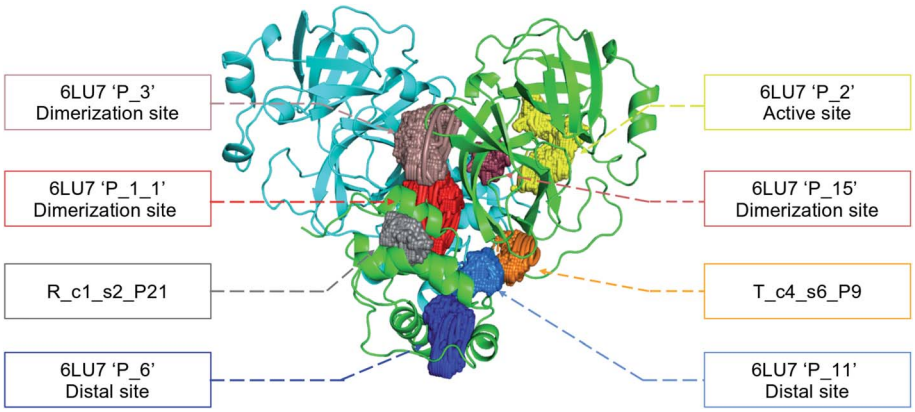


Fig. 9 Representation of the pocket locations on the 6LU7 SARS-CoV-2 main protease structure.

'R_c1_s18_P14' or 'T_c4_s19_P3' or (2) in many structures, such as 'R_c1_s2_P20', 'R_c1_s2_P25' or 'R_c1_s4_P7'. Interestingly, only 3 pockets were retrieved in clusters 4 and 5 of the Tinker-HP simulations: 'T_c4_s2_P8', 'T_c4_s5_P5' and 'T_c4_s6_P9'. The last one, 'T_c4_s6_P9' is of particular interest since its volume is equal to 199 Å^3 and its druggability score, DrugScore,[65] reaches 0.62. We repeated the pocket detection and analysis procedure on 100 randomly selected structures (20 for each of the 5 clusters) identified within the Tinker-HP simulations (see Fig. 10 in the ESI†). We observed that the 3 previously identified pockets 'T_c4_s2_P8', 'T_c4_s5_P5' and 'T_c4_s6_P9' were also detected on the structures randomly selected in clusters 4 and 5 of the Tinker HP simulations but also partially in cluster 3. We then evaluated if all the pockets assigned to the 'T_c4_s6_P9' pocket displayed similar properties. We observed that the mean volume of these pockets was 215 Å^3 but few structures presented extreme values far superior to this mean volume (Fig. 12 in the ESI†). Similarly, the DrugScore mean value was 0.37 but with large variations among the structures and the clusters (see Fig. 13 in the ESI†). For comparison, we also computed the DrugScore value distribution for each newly identified pocket, *i.e.* pockets that were not detected in the 6LU7 structure (Fig. 14 in the ESI†). One pocket, 'R_c1_s2_P21', displays peculiar properties with a mean druggability value of 0.6 and a mean volume value of 150 Å^3 which seems to indicate that this pocket may only accommodate very small compounds. The discovery of the 'T_c4_s6_P9' pocket is thus a very promising result, but one that underlines the necessity of carefully selecting one or several structure(s) in which the pocket properties are optimal for further *in silico* investigations to identify small molecules able to modulate the SARS-CoV-2 protease activity. All the pockets discussed herein are represented within the 6LU7 structure in Fig. 9.

6 Solvation analysis: the importance of including explicit polarization effects in water

Water molecules play critical roles in enzyme and protein functioning. In fact water can be a product or a reactant in condensation and hydrolysis reactions, a transition state intermediate in chemical reactions and a structural element at the molecular level. In the lattermost case, water interconnects the protein through hydrogen bonds in order to maintain and stabilize the positions of the residues and the fold.[66] Previous experimental studies on SARS-CoV-1 and SARS-CoV-2 have shown that one structural water molecule was conserved within the main protease of the two viruses and interacts with the cyclic nitrogen of His41.[38,51,52] A recent crystallographic study on SARS-CoV-2 suggests that another water molecule could be observed around His163.[49] In order to calculate the number of water molecules inside the active site and in proximity of His41 and His163 of both protomers, we have created a virtual sphere of 4 Å, centered on the nitrogen of each of the two concerned histidines and have calculated the number of water molecules inside the active site of each protomer over time. Fig. 11 shows the dipole distribution of structural water molecules for protomers 1 and 2 of His163 (a and b) and His41 (c and d). The AMOEBA results are striking. They show that (i) the water molecules in each of the two protomers' active sites are highly polarized, and (ii) the AMOEBA distribution of the water molecules is significantly different from the ones observed in

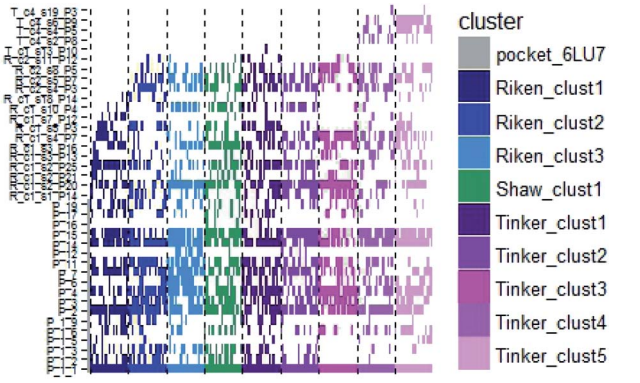


Fig. 10 Schematic representation of the detected DoGSite Score pockets within the 6LU7 structure (first column on the left, represented in grey) and 20 structures extracted from each cluster identified within RIKEN (blue gradient), DESRES (green) and Tinker-HP (magenta gradient) simulations.

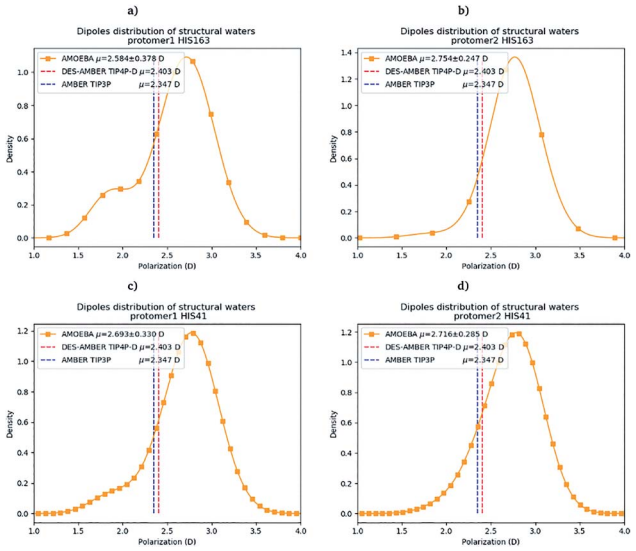


Fig. 11 Dipole distribution of water molecules for protomers 1 and 2 around His163 (a and b) and around His41 (c and d).

the DESRES TIP4-D (DES-AMBER) and RIKEN TIP3P (AMBER) trajectories. High polarization has been shown in past studies to be a common feature of structural water molecules that exhibit high dipole moments.[67] In practice, the average dipole moment having the highest density with the AMOEBA force field is located around 2.9 D while for the DES-AMBER and AMBER n-PFFs, the water dipoles are fixed at 2.403 D and 2.347 D, respectively (see Fig. 5). Since AMOEBA dipole moments are not fixed, we observe strong polarization fluctuations due to water traffic inside the catalytic region. Fig. 15 in the ESI† presents the number of structural water molecules for protomers 1 and 2 of His163 (a and b) and His41 (c and d). All trajectories show a highest density for no water molecules within a distance of 4 Å from protomer 1 of His163. However, this observation is different for protomer 1 of His41 where Tinker-HP trajectories found a highest density for the presence of one water molecule while it was 2 molecules for RIKEN's and 4 molecules for DESRES's trajectories. A non-symmetric distribution of water molecules compared to protomer 1 is found for protomer 2. Tinker-HP and RIKEN trajectories do not predict the frequent presence of water molecules within the chosen distance from His163, while DESRES's trajectories exhibit a higher density for 1 molecule. Concerning His41 of protomer 2, Tinker-HP's and DESRES's trajectories show a most frequent density of one water molecule, while RIKEN's highest density goes to 2 water molecules, and slightly less for 1 molecule. These observations demonstrate that water polarization intensively fluctuates inside the confined active site, suggesting a dynamic role of polarization on water traffic that strongly influences water molecule interactions with His163 and His41 of each of the two protomers. However these interactions are not distributed symmetrically between protomers. So is it compatible with experimental data? Again, relatively detailed X-ray data exist for other coronaviruses including SARS-CoV-1 where the role of histidines has been extensively discussed.[51,52] The presence of a structural water molecule around His41 is always confirmed. For SARS-CoV-2, papers describing the M^{pro} protease structure in its apo state[38,49] under physiological pH conditions also discuss the presence of such molecule found near the catalytic dyad (His41). However, the interaction of the structural water molecule with His163 appears to only be proposed in Zhou *et al.*'s report.[49]

Concerning the precise predicted water count around His41, AMBER and DES-AMBER have on average a higher number of structural water molecules (2.38 to 4.01 at the most) compared to AMOEBA which predicts the presence of 1.5 water molecules, more in line with accumulated experimental data. Fig. 15 in the ESI† shows that the non-polarizable simulations capture frequent configurations with up to 4 water molecules which could be a consequence of the non-inclusion of the polarization effect leading to a weaker and constant dipole moment of the water molecules that could generate more water traffic. Compared to His41, all AMOEBA, AMBER, and DES-AMBER analyses found significantly fewer water molecules around His163. In practice AMOEBA found the lowest water count of all methods with an average of 0.13–0.31 molecules around His163, while the higher trends observed for His41 are still present for all n-PFFs except for one protomer of DES-AMBER that exhibits 0.77 molecules (see Table 2). Clearly, the presence of a structural water molecule around His163 seems less probable for all simulations (under the present pH conditions) and in competition with the water traffic entering the measurement sphere. The dipole distribution of water molecules offers further analysis as it is found to be slightly larger for His163 and associated with a smaller density of highly polarized total dipole moments confirming the trends. In any case, the presence of water in the active site thus appears consistent with the need for a water molecule to model the enzyme reaction mechanism.[38,68]

Table 2 Average and standard deviation of the number of water molecules around His163 and His41 residues using AMOEBA for simulations at pH 7.4 and 6

	His163		His41	
	Protomer 1	Protomer 2	Protomer 1	Protomer 2
AMOEBA pH 6	0.37, $\sigma = 0.65$	0.27, $\sigma = 0.57$	1.95, $\sigma = 1.04$	1.42, $\sigma = 0.97$
AMOEBA pH 7.4	0.31, $\sigma = 0.51$	0.13, $\sigma = 0.34$	1.48, $\sigma = 0.99$	1.62, $\sigma = 1.06$
Experiments	0 or 1		1	

7 Further simulation at lower pH: impact of His172 protonation

From the past studies on SARS-CoV-1 (see ref. 51 and references therein) we know that the activity of the main protease system is pH dependent. While its activity is lower at low pH and high pH, it is higher at pH close to the physiological human pH (*i.e.* 7.4). Studies performed on the M^{pro} of SARS-CoV-1 show a bell-shaped pH–activity curve[51] for the enzyme. All proposed simulations (*i.e.* ours and the one from DESRES and RIKEN) were performed using neutral histidine residues. Indeed, one key element of the impact of lowering the pH is the protonation of His172 and His163.[51] Initially, based on SARS-CoV-1 knowledge, it was thought that if His172 and His163 were not protonated at pH = 8, His172 would be in a protonated state in both protomers at physiological pH (pH = 7.4) since its pK_a was found to be close to 7.6.[69] However, differences exist with the SARS-CoV-2 M^{pro}, and Verma *et al.* recently showed[37] that the pK_a of His172 would be actually lower than anticipated, being about 6.6. Such prediction appears consistent with recent experimental results.[38] Our proposed simulation setup using neutral histidines is therefore likely to be consistent with physiological pH conditions. In that connection, Verma *et al.* described the critical role of the protonation of His172 on the holo state that would happen at pH = 6 and they showed that it would lead to a partial collapse of the S1 pocket, linked with a strong destructuring of the oxyanion hole.[37] Thus, it appears critical to investigate the influence of pH on our apo results by performing an additional simulation compatible with pH = 6 conditions. So, in order to propose a starting point for this second simulation, we followed a protocol found in the literature for SARS-CoV-1.[51] We then selected 15 new structures from our pH = 7.4 simulation (3 structures per cluster). For each structure we then protonated the His172 on both protomers, which initiates the structural transformation from pH = 7.4 to pH = 6. The same simulation protocol (see Section 3.2) was followed and a total of 17 μs of simulation was thus generated using the Jean Zay Supercomputer (IDRIS, GENCI, France). In practice, with enough sampling, the structures should be able to relax. Of course, as pointed out by Verma *et al.*,[37] other residues could be impacted by lowering the pH but such simulation has strong interpretative interest. We therefore looked again at all the structural markers described for the previous simulation. We first studied the convergence of some of the properties. Fig. 11 in the ESI† shows that the simulation tends to converge more slowly than at physiological pH and starts to do so beyond 14 μs. Clearly, comparisons of both pH situations would not have been possible using nanosecond simulations even if initial local relaxation of the histidine residues appears to have happened at this timescale. Of course, we cannot state that the simulation is fully converged. However, we stopped the computation when the observed structural changes strongly diminished over time within the ensemble, leaving us with enough confidence in the computed properties. The key result obtained from this second long simulation is the strong variation of the activation features present in the previously described inactive protomer. Indeed, while a significant asymmetry between protomers was found at pH = 6 with protomer 1 exhibiting a poor structure oxyanion hole, the situation evolves with the protonation of His172. Indeed protomer 1 now exhibits a mix of several states with different structural markers (see Fig. 16, ESI†). Compared to pH = 7.4, the interaction of His172/163 with Glu166 changed from a H-bond type interaction (neutral His172/163 at pH = 7.4) to a salt-bridge (positively charged His172 at pH = 6).[70] The stacking index shows that the stacking interaction appears to be weaker than at physiological pH and therefore easier to break and to form (see ESI Fig. 17†). As a result of the protonation, protomer 1 now shows two relatively short maxima for the Glu166–His172 distance (see ESI Fig. 16†) associated with a continuum of values of distances going beyond 6 Å. The protomer 1 Glu166–His172 distance appears to explore a variety of situations including a favorable stacking second minimum which is a sign of a more structured state. However, while some ordered states are found, the absence of stacking is statistically dominant and associated with a striking set of Glu166–His163 interactions. Clearly some really short hydrogen-bonds are found between these residues, a sign of a strong destructuring of the oxyanion hole. These results are in line with the findings of Verma *et al.*[37] that associated the protonation of His172 with the collapse of the oxyanion loop toward the S1 pocket. However, for the other protomer, our apo results differ a bit from Verma *et al.*'s holo data. Indeed, the situation appears more contrasted. Despite a net destructuring effect, protomer 2 tends also to exhibit a mix of states after protonation. The protomer encompasses longer Glu166–His172 interactions than previously noted at physiological pH and the noticeable appearance of some states with short Glu166–His163 distances is observed. However, in the case of protomer 2, the stacking still statistically partially holds despite the existence of a second peak describing a non-negligible absence of stacking in some configurations. Overall, our computations show that the protomers tend to be both affected by the destructuring effect of the His172 protonation, leading to a more symmetrical situation between destructured protomers. Protonation of His172 definitively increases the dynamical aspect of the protease structure and favors the exploration of different states of the activation markers highlighting the instability of the oxyanion hole leading to the partial collapse of the S1 pocket. The impact of the increased flexibility can be further examined through the comparative RMSF of the two simulated pH states where the mobility of the C-terminal end appears further enhanced (see ESI Fig. 17†). This clearly correlates with our initial remark concerning the sampling, that such lower pH structure is far more complex to simulate than the situation at physiological pH as several states resonate due to the low structuring of the oxyanion loop. Finally, Table 2 shows the evolution of the solvation around His163 and His41. The number of water molecules found in the AMOEBA simulation tends to increase on both histidine sites compared to pH = 7.4 with more configurations including one and two water molecules for His163 and His141, respectively. If the presence of a structural water molecule is confirmed around His41, a similar presence around His163 tends to be statistically reinforced under these

protonation conditions. Clearly these findings have potentially an important impact in drug discovery as the presence of structural water molecules around His141 and potentially His163 would make rational drug design more difficult since the substrate or inhibitors would suffer from steric hindrance.[49] The use of PFFs could be critical in the evaluation of the free energies of binding of possible drug candidates. Indeed, our data confirm the high plasticity of the active site observed in X-ray structures[38] at room temperature. Modeling such plasticity including the structuring of the S1 pocket clearly requires the simultaneous capability to accurately evaluate various types of weak interaction including hydrogen bonds, salt bridges and π–π stacking while high-resolution modeling of solvation appears to also be mandatory. Of course, we also showed that extensive sampling beyond the μs-timescale was crucial to deal with such difficult flexible systems.

8 Conclusion and perspectives

In this work, designed in response to the urgent need for COVID-19 research, we demonstrated that it is now possible to perform long μs-timescale MD simulations of large biosystems using polarizable force fields such as AMOEBA that are able to account for physical many-body effects. Due to the inherent complexity of the SARS-CoV-2 proteins, performing such higher-resolution simulations is important as they could provide additional information about the structural dynamics of virus constituents to the COVID-19 experimental and computational research communities. To do so, we proposed a fully unsupervised adaptive sampling strategy that can be used on any type of computational resources. This automated framework allows for production simulations that benefit from advances in supercomputing and from our recent Tinker-HP HPC massively parallel software enhancements, that can now efficiently handle GPU-accelerated large petascale computers using lower precision arithmetic and MPI. In order to extract new information from this type of simulation, we also provided the necessary steps to remove the bias from (re-weight) the obtained data to collect useful and accurate structural dynamics features. More than 38 μs of all-atom MD simulation of the M^{pro} enzyme in its apo (ligand-free) state was produced using the AMOEBA polarizable force field.

Results were then compared to available state-of-the-art large scale simulation data. The results from the new generation PFF were shown to capture most of the structural dynamics features discussed in the experimental literature, confirming that M^{pro} is probably in a poorly active conformation in its apo state under physiological pH conditions. However, simulations detected some partial activity features in one of the protomers linked to a more structured oxyanion hole. This is consistent with the protomeric asymmetric activity observed in the holo state where only one protomer is found to be active,[48] a similar feature that was also observed in SARS-CoV-1.[54] This asymmetry can be related to several structural markers as well as to the total protomer volumes. The active site is found to be highly flexible at room temperature in agreement with recent experimental findings.[38] Overall, the apo state of M^{pro} clearly appears less organized than the holo state in agreement with experimental results discussed by Zhou *et al.*[49] A second simulation, including the protonation of the His172 residue to simulate the system under pH = 6 conditions, was performed and tends to confirm the role of the protonation in the collapse of the S1 pocket at lower pH. Under these conditions, the protomeric AMOEBA asymmetry remains although the protomers tend to be notably destructured. The AMOEBA simulations also captured the C-terminal high flexibility feature discussed in the literature.[49] Flexibility increases at lower pH and tends to further modulate down the activity of the apo state linked with the collapse of the S1 pocket. Striking differences were observed concerning the solvation patterns around the key His41 and His163 residues between AMOEBA and n-PFFs. Overall, the smaller AMOEBA water count around histidines is more in line with experimental data. If the presence of a structural water molecule around His41 is probable at all pH, the existence of a water molecule around His163 tends to be more statistically possible at pH = 6. These results can be explained by the capability of AMOEBA structural water molecules to exhibit an average dipole moment higher than that of bulk water and to explore a wider range of dipoles compared to n-PFFs. Structural water molecules around histidines will clearly affect rational drug design. The use of polarizable force fields could be critical in the evaluation of the free energies of binding of possible drug candidates competing with water to interact with the enzyme. In practice, the M^{pro} enzyme tends to be difficult for molecular mechanics approaches. Indeed, it encompasses all sorts of weak interactions. Therefore, it is not surprising that all the experimentally described features found within the AMOEBA simulations were not necessarily found with the non-polarizable simulations. Such systems tend to require both an accurate force field and an extensive sampling strategy as it is obvious that a few ns of PFF MD alone would not provide insights into a system where the statistical convergence is challenging due to its plasticity. These results provide a first direct validation of the stability of the AMOEBA polarizable force field and clearly demonstrate its applicability at long timescales. Besides correlating with experimental data, our results also show that our adaptive sampling approach coupled with AMOEBA led to enhanced volumes for the active site and to additional potential cryptic pockets as well. As the apo (ligand-free) state has been shown to be a relevant structure at room temperature to perform docking studies,[38] the new information provided could be useful for drug design. Our simulation data are fully available to the general public. They can therefore be used for further structural analysis and/or as an additional basis for ensemble docking studies.[71] Indeed, concentrating the GPU computing power on an apo state is useful to "mine" the conformations to obtain an accurate and more statistically converged set of MD binding site conformations that could be selected by a ligand. The new structural information provided here could help to design new drugs or to repurpose existing ones. These data could also be important to understand chemical reactivity at an atomic level *via* hybrid QM/MM simulations.[68,72] Finally, thanks to the presented divide and conquer strategy, our AMOEBA adaptive MD simulations were

shown to be simultaneously computationally competitive and in line with the available experimental data. Using 100 GPU cards, we show that an acceptable and competitive time to solution could be achieved as our "microsecond" results were obtained in a few days on an academic (and multipurpose) supercomputer. It is worth noting that each simulation could have run on full nodes or using more efficient A100 cards. In practice, a similar exploration of the available community data was already achieved in only 2.5 days (Fig. 1). It is also important to note that Tinker-HP can also produce an order of magnitude faster simulation using n-PFFs using GPUs. Since n-PFF simulations are also of great interest, capturing many experimental aspects, our dual-level (n-PFF + PFF) strategy is confirmed. Indeed, an optimal setup consists in first producing a long adaptive non-polarizable simulation that can be further refined with polarizable potentials within additional adaptive iterations. That way, our approach could also use Folding@-home COVID-19 community results[73] as an input (or any available data shared on the BioExcel/Molssi repository) in order to deliver a maximum of potentially new/useful information into COVID-19 research. Indeed, it is important to recall the importance of proposing accurate (and as much as possible converged) simulations of the COVID-19 targets. As a final perspective, we can mention that the present strategy is platform independent and not limited to supercomputers. Therefore, it can also be used at a smaller scale on "cheaper" laboratory GPU clusters which can benefit from the computational power of low arithmetic to obtain local supercomputing capabilities. On the other side of the spectrum, with the coming of the exascale era and the HPC–Artificial Intelligence (AI) convergence, the "big iron" supercomputer systems, and their cloud-computing counterparts, will considerably extend the high accuracy conformational mining capabilities leading to extended possibilities for the *in silico* modeling of complex biological systems.

Author contributions

T. J. I., F. C., D. El A., and N. L. performed simulations; O. A., T. J. I., and L.-H. J. contributed new code. P. M., T. J. I., J.-P. P, P. R., and L. L. contributed new methodology. N. L., M. M., L. L., F. C., and P. M. contributed analytical tools. F. C., T. J. I., D. El A., N. L., M. M., P. R., and J.-P. P. analyzed data. J.-P. P., P. M., L. L., N. L., T. J. I., F. C. and P. R. wrote the paper; J.-P. P. designed the research.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement no. 810367), project EMC2 (JPP). FC acknowledges funding from the French state funds managed by the CalSimLab LABEX and the ANR within the Investissements d'Avenir program (reference ANR11-IDEX-0004-02) and support from the Direction Génerale de l'Armement (DGA) Maîtrise NRBC of the French Ministry of Defense. DEA acknowledges funding from the Lebanese National Council for Scientific Research, CNRS-L. Adaptive sampling computations were performed at GENCI thanks to a COVID19 emergency allocation on the Jean Zay machine (IDRIS, Orsay, France) under grant no. A0070707671 and on the Irene Joliot Curie machine thanks to a PRACE COVID-19 emergency grant (project COVID-HP). The authors thank the Swiss National Supercomputing Center (CSCS) for hosting our data through the FENIX infrastructure. JPP acknowledges a special COVID-19 funding from Sorbonne Université. PR is grateful for support by the Robert A. Welch Foundation (F-1691) and National Institutes of Health (R01GM106137 and R01GM114237).

References

1 J. Guarner, *Am. J. Clin. Pathol.*, 2020, **153**, 420–421.
2 F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei, *et al.*, *Nature*, 2020, **579**, 265–269.
3 Z. Jin, X. Du, Y. Xu, Y. Deng, M. Liu, Y. Zhao, B. Zhang, X. Li, L. Zhang, C. Peng, *et al.*, *Nature*, 2020, 1–5.
4 D. Leung, G. Abbenante and D. P. Fairlie, *J. Med. Chem.*, 2000, **43**, 305–341.
5 T. S. Komatsu, Y. Koyama, N. Okimoto, G. Morimoto, Y. Ohno and M. Taiji, Mendeley Data, 2020, DOI: 10.17632/vpps4vhryg.2.
6 *DESRES: Molecular Dynamics Simulations Related to SARS-CoV-2*, 2020, DESRES-ANTON-10880334.
7 M. M. Ghahremanpour, J. Tirado-Rives, M. Deshmukh, J. A. Ippolito, C.-H. Zhang, I. C. de Vaca, M.-E. Liosi, K. S. Anderson and W. L. Jorgensen, *ACS Med. Chem. Lett.*, 2020, **11**(12), 2526–2533.
8 T. Sztain, R. Amaro and J. A. McCammon, *bioRxiv*, 2020, DOI: 10.1101/2020.07.23.218784.
9 S. Piana, P. Robustelli, D. Tan, S. Chen and D. E. Shaw, *J. Chem. Theory Comput.*, 2020, **16**, 2494–2507.
10 J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.
11 D. E. Shaw, J. P. Grossman, J. A. Bank, B. Batson, J. A. Butts, J. C. Chao, M. M. Deneroff, R. O. Dror, A. Even, C. H. Fenton, A. Forte, J. Gagliardo, G. Gill, B. Greskamp, C. R. Ho, D. J. Ierardi, L. Iserovich, J. S. Kuskin, R. H. Larson, T. Layman, L. Lee, A. K. Lerer, C. Li, D. Killebrew, K. M. Mackenzie, S. Y. Mok, M. A. Moraes, R. Mueller, L. J. Nociolo, J. L. Peticolas, T. Quan, D. Ramot, J. K. Salmon, D. P. Scarpazza, U. B. Schafer, N. Siddique, C. W. Snyder, J. Spengler, P. T. P. Tang, M. Theobald, H. Toma, B. Towles, B. Vitale, S. C. Wang and C. Young, *SC'14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2014, pp. 41–53.

12 I. Ohmura, G. Morimoto, Y. Ohno, A. Hasegawa and M. Taiji, *Philos. Trans. R. Soc., A*, 2004, **372**, 20130387.
13 Y. Shi, P. Ren, M. Schnieders and J.-P. Piquemal, Polarizable force fields for biomolecular modeling, in *Reviews in Computational Chemistry*, ed. A. L. Parrill and K. B. Lipkowitz, John Wiley and Sons, Inc., Hoboken, NJ, 2015, vol. 28, pp. 51–86, DOI: 10.1002/9781118889886.ch2.
14 Z. Jing, C. Liu, S. Y. Cheng, R. Qi, B. D. Walker, J.-P. Piquemal and P. Ren, *Annu. Rev. Biophys.*, 2019, **48**, 371–394.
15 J. Melcr and J.-P. Piquemal, *Front. Mol. Biosci.*, 2019, **6**, 143.
16 F. Célerse, L. Lagardère, E. Derat and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2019, **15**, 3694–3709.
17 L. El Khoury, F. Célerse, L. Lagardère, L.-H. Jolly, E. Derat, Z. Hobaika, R. G. Maroun, P. Ren, S. Bouaziz, N. Gresh, *et al.*, *J. Chem. Theory Comput.*, 2020, **16**, 2013–2020.
18 *GENCI: lutte contre le COVID-19*, online https://www.genci.fr/fr/content/projets-contre-le-covid-19, 2020.
19 *European PRACE Support to Mitigate Impact of COVID-19 Pandemic*, https://prace-ri.eu/prace-support-to-mitigate-impact-of-covid-19-pandemic/, 2020.
20 *United States COVID-19 High Performance Computing Consortium*, https://covid19-hpc-consortium.org/, 2020.
21 L. Lagardère, L.-H. Jolly, F. Lipparini, F. Aviat, B. Stamm, Z. F. Jing, M. Harger, H. Torabifard, G. A. Cisneros, M. J. Schnieders, N. Gresh, Y. Maday, P. Y. Ren, J. W. Ponder and J.-P. Piquemal, *Chem. Sci.*, 2018, **9**, 956–972.
22 O. Adjoua, L. Lagardère, L.-H. Jolly, A. Durocher, Z. Wang, T. Very, I. Dupays, F. Célerse, J. Ponder, P. Ren and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2021, arXiv: 2011.01207.
23 G. R. Bowman, D. L. Ensign and V. S. Pande, *J. Chem. Theory Comput.*, 2010, **6**, 787–794.
24 M. I. Zimmerman, J. R. Porter, X. Sun, R. R. Silva and G. R. Bowman, *J. Chem. Theory Comput.*, 2018, **14**, 5459–5475.
25 R. M. Betz and R. O. Dror, *J. Chem. Theory Comput.*, 2019, **15**, 2053–2063.
26 E. Hruska, J. R. Abella, F. Nüske, L. E. Kavraki and C. Clementi, *J. Chem. Phys.*, 2018, **149**, 244119.
27 L.-H. Jolly, A. Duran, L. Lagardère, J. W. Ponder, P. Ren and J.-P. Piquemal, *LiveCoMS*, 2019, **1**, 10409.
28 H. Abdi and L. J. Williams, *Wiley Interdiscip. Rev. Comput. Stat.*, 2010, **2**, 433–459.
29 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, *J. Mach. Learn. Res.*, 2011, **12**, 2825–2830.
30 R. T. McGibbon, K. A. Beauchamp, M. P. Harrigan, C. Klein, J. M. Swails, C. X. Hernández, C. R. Schwantes, L.-P. Wang, T. J. Lane and V. S. Pande, *Biophys. J.*, 2015, **109**, 1528–1532.
31 P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt and SciPy 1.0 Contributors, *Nat. Methods*, 2020, **17**, 261–272.
32 A. Laio and M. Parrinello, *Proc. Natl. Acad. Sci. U. S. A.*, 2002, **99**, 12562–12566.
33 P. Y. Ren and J. W. Ponder, *J. Phys. Chem.*, 2003, **107**, 5933–5947.
34 Y. Shi, Z. Xia, J. Zhang, R. Best, C. Wu, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2013, **9**, 4046–4063.
35 J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. A. DiStasio, M. Head-Gordon, G. N. I. Clark, M. E. Johnson and T. Head-Gordon, *J. Phys. Chem. B*, 2010, **114**, 2549–2564.
36 C. Zhang, C. Lu, Z. Jing, C. Wu, J.-P. Piquemal, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2018, **14**, 2084–2108.
37 N. Verma, J. A. Henderson and J. Shen, *J. Am. Chem. Soc.*, 2020, **142**, 21883–21890.
38 D. Kneller, G. Phillips, H. O'Neill, R. Jedrzejczak, L. Stols, P. Langan, A. Joachimiak, L. Coates and A. Kovalevsky, Structural plasticity of SARS-CoV-2 3CL M^{pro} active site cavity revealed by room temperature X-ray crystallography, *Nat. Commun.*, 2020, **11**, 3202.
39 J. A. Rackers, Z. Wang, C. Lu, M. L. Laury, L. Lagardère, M. J. Schnieders, J.-P. Piquemal, P. Ren and J. W. Ponder, *J. Chem. Theory Comput.*, 2018, **14**, 5273–5289.
40 L. Lagardère, F. Aviat and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2019, **10**, 2593–2599.
41 *Data Tinker-HP*, SARS–CoV–2 Main Protease, deposited at CSCS, 2020.
42 A. Amadei, A. B. Linssen and H. J. Berendsen, *Proteins*, 1993, **17**, 412–425.
43 A. Amadei, A. Linssen, B. De Groot, D. Van Aalten and H. Berendsen, *J. Biomol. Struct. Dyn.*, 1996, **13**, 615–625.
44 H. J. Berendsen and S. Hayward, *Curr. Opin. Struct. Biol.*, 2000, **10**, 165–169.
45 M. Ester, H.-P. Kriegel, J. Sander and X. Xu, *et al.*, *Kdd*, 1996, pp. 226–231.
46 Y. Liu, Z. Li, H. Xiong, X. Gao and J. Wu, *2010 IEEE International Conference on Data Mining*, 2010, pp. 911–916.
47 C. D. Owen, P. Lukacik, C. M. Strain-Damerell, A. Douangamath, A. J. Powell, D. Fearon, J. Brandao-Neto, A. D. Crawshaw, D. Aragao, M. Williams, R. Flaig, D. Hall, K. McAauley, D. I. F. Stuartvon Delft and M. A. Walsh, PDB 6Y84: Structure COVID-19 main protease with unliganded active site, 2020, https://www.wwpdb.org/.
48 L. Zhang, D. Lin, X. Sun, U. Curth, C. Drosten, L. Sauerhering, S. Becker, K. Rox and R. Hilgenfeld, *Science*, 2020, **368**, 409–412.
49 X. Zhou, F. Zhong, C. Lin, X. Hu, Y. Zhang, B. Xiong, X. Yin, J. Fu, W. He, J. Duan, *et al.*, *Sci. China: Life Sci.*, 2020, 1–4.
50 H. Yang, M. Yang, Y. Ding, Y. Liu, Z. Lou, Z. Zhou, L. Sun, L. Mo, S. Ye, H. Pang, *et al.*, *Proc. Natl. Acad. Sci. U. S. A.*, 2003, **100**, 13190–13195.

51 J. Tan, K. H. Verschueren, K. Anand, J. Shen, M. Yang, Y. Xu, Z. Rao, J. Bigalke, B. Heisen, J. R. Mesters, K. Chen, X. Shen, H. Jiang and R. Hilgenfeld, *J. Mol. Biol.*, 2005, **354**, 25–40.
52 H. Yang, M. Yang, Y. Ding, Y. Liu, Z. Lou, Z. Zhou, L. Sun, L. Mo, S. Ye, H. Pang, G. F. Gao, K. Anand, M. Bartlam, R. Hilgenfeld and Z. Rao, *Proc. Natl. Acad. Sci. U. S. A.*, 2003, **100**, 13190–13195.
53 L. Zhang, D. Lin, X. Sun, U. Curth, C. Drosten, L. Sauerhering, S. Becker, K. Rox and R. Hilgenfeld, *Science*, 2020, **368**, 409–412.
54 H. Chen, P. Wei, C. Huang, L. Tan, Y. Liu and L. Lai, *J. Biol. Chem.*, 2006, **281**, 13894–13898.
55 D. Branduardi, F. L. Gervasio, A. Cavalli, M. Recanatini and M. Parrinello, *J. Am. Chem. Soc.*, 2005, **127**, 9147–9155.
56 J. Hermans, in *Peptide Solvation and HBonds*, Academic Press, 2005, vol. 72, Advances in Protein Chemistry, pp. 105–119.
57 R. S. Paton and J. M. Goodman, *J. Chem. Inf. Model.*, 2009, **49**, 944–955.
58 J. A. Lemkul, J. Huang, B. Roux and A. D. MacKerell, *Chem. Rev.*, 2016, **116**, 4983–5013.
59 S. Cardamone, T. J. Hughes and P. L. A. Popelier, *Phys. Chem. Chem. Phys.*, 2014, **16**, 10367–10387.
60 B. Goyal and D. Goyal, *ACS Comb. Sci.*, 2020, **22**, 297–305.
61 J. Liang, C. Karagiannis, E. Pitsillou, K. K. Darmawan, K. Ng, A. Hung and T. C. Karagiannis, *Comput. Biol. Chem.*, 2020, 107372.
62 J. R. Wagner, J. Sørensen, N. Hensley, C. Wong, C. Zhu, T. Perison and R. E. Amaro, *J. Chem. Theory Comput.*, 2017, **13**, 4584–4592.
63 A. Volkamer, D. Kuhn, F. Rippmann and M. Rarey, *Bioinformatics*, 2012, **28**, 2074–2075.
64 B. Goyal and D. Goyal, *ACS Comb. Sci.*, 2020, **22**, 297–305.
65 P. Schmidtke and X. Barril, *J. Med. Chem.*, 2010, **53**, 5858–5867.
66 Y. Levy and J. N. Onuchic, *Annu. Rev. Biophys. Biomol. Struct.*, 2006, **35**, 389–415.
67 B. de Courcy, J.-P. Piquemal, C. Garbay and N. Gresh, *J. Am. Chem. Soc.*, 2010, **132**, 3312–3320.
68 K. Świderek and V. Moliner, *Chem. Sci.*, 2020, **11**, 10626–10630.
69 J. Yang, M. Yu, Y. N. Jan and L. Y. Jan, *Proc. Natl. Acad. Sci. U. S. A.*, 1997, **94**, 1568–1572.
70 S.-M. Liao, Q.-S. Du, J.-Z. Meng, Z.-W. Pang and R.-B. Huang, *Chem. Cent. J.*, 2013, **7**, 44.
71 R. E. Amaro, J. Baudry, J. Chodera, Ö. Demir, J. A. McCammon, Y. Miao and J. C. Smith, *Biophys. J.*, 2018, **114**, 2271–2278.
72 D. Loco, L. Lagardère, G. A. Cisneros, G. Scalmani, M. Frisch, F. Lipparini, B. Mennucci and J.-P. Piquemal, *Chem. Sci.*, 2019, **10**, 7200–7211.
73 M. I. Zimmerman, J. R. Porter, M. D. Ward, S. Singh, N. Vithani, A. Meller, U. L. Mallimadugula, C. E. Kuhn, J. H. Borowsky, R. P. Wiewiora, M. F. D. Hurley, A. M. Harbison, C. A. Fogarty, J. E. Coffland, E. Fadda, V. A. Voelz, J. D. Chodera and G. R. Bowman, *bioRxiv*, 2020, DOI: 10.1101/2020.06.27.175430.

Conclusion

The unsupervised data-driven adaptive sampling framework can efficiently manage GPU-accelerated large petascale computers allowing to generate more than 50 μs of all-atom MD simulation for the M^{pro} enzyme using the AMOEBA PFF, which enabled high-resolution exploration of its conformational space. The resulting simulations were compared with state-of-the-art large-scale simulation data, revealing that the PFF was able to capture most of the structural dynamics features discussed in the experimental literature, unlike nPFF. The accuracy of the force field and extensive sampling strategy is critical for such systems. These results provide direct validation of the stability of the AMOEBA PFF and demonstrate its applicability at long timescales. Additionally, the adaptive sampling approach coupled with AMOEBA led to enhanced volumes for the active site and potential additional cryptic pockets.

The μs simulations were obtained in just a few days on an academic supercomputer, demonstrating that an acceptable and competitive time to solution can be achieved with this technique. Furthermore, the adaptive sampling strategy is platform-independent and does not require prior structural insight into the protein since it is unsupervised.

3.2 Exploring Water-Driven Allosteric Interactions of SARS-CoV-2 M^{pro} through Adaptive Sampling

Introduction

This section presents a more detailed analysis of the extensive 50 μs simulations of M^{pro} using AMOEBA and GPUs-accelerated unsupervised adaptive sampling strategy, along with 100 μs simulations with nPFF, which were explained in the previous section. This section presents a detailed analysis of the extensive 50 μs simulations of M^{pro} using AMOEBA and GPUs-accelerated unsupervised adaptive sampling strategy, along with 100 μs simulations with nPFF, which were explained in the previous section. Notably, significant differences in structural dynamics were observed in key parts of M^{pro} compared to nPFFs. The current study is focused on the factors that structure the dimerization interface as a function of different pH and FF models. Specifically, we investigate the role of many-body effects in the modeling of interfacial water. [172]

THE JOURNAL OF PHYSICAL CHEMISTRY LETTERS
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

pubs.acs.org/JPCL

Letter

Interfacial Water Many-Body Effects Drive Structural Dynamics and Allosteric Interactions in SARS-CoV-2 Main Protease Dimerization Interface

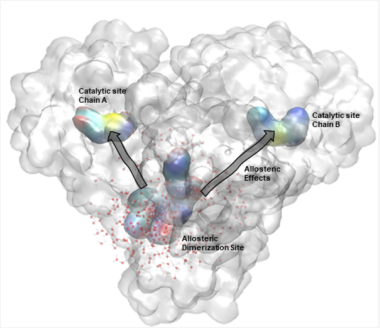
Dina El Ahdab, Louis Lagardère, Théo Jaffrelot Inizan, Fréderic Célerse, Chengwen Liu, Olivier Adjoua, Luc-Henri Jolly, Nohad Gresh, Zeina Hobaika, Pengyu Ren, Richard G. Maroun, and Jean-Philip Piquemal*

Cite This: J. Phys. Chem. Lett. 2021, 12, 6218−6226

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Following our previous work (*Chem. Sci.* 2021, *12*, 4889−4907), we study the structural dynamics of the SARS-CoV-2 Main Protease dimerization interface (apo dimer) by means of microsecond adaptive sampling molecular dynamics simulations (50 μs) using the AMOEBA polarizable force field (PFF). This interface is structured by a complex H-bond network that is stable only at physiological pH. Structural correlations analysis between its residues and the catalytic site confirms the presence of a buried allosteric site. However, noticeable differences in allosteric connectivity are observed between PFFs and non-PFFs. Interfacial polarizable water molecules are shown to appear at the heart of this discrepancy because they are connected to the global interface H-bond network and able to adapt their dipole moment (and dynamics) to their diverse local physicochemical microenvironments. The water−interface many-body interactions appear to drive the interface volume fluctuations and to therefore mediate the allosteric interactions with the catalytic cavity.



In the context of COVID-19 drug discovery, both structural and nonstructural proteins are considered as promising targets for the development of antiviral agents against the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2).[1] Specifically, SARS-CoV-2 M^{pro} plays a pivotal role in controlling viral replication and transcription through proteolytic processing of viral poly proteins.[2] Many studies on inhibitor ligands are based on active site pocket targeting. However, advancing a drug toward clinical trials remains a daunting task[3] (as was the case for SARS-Cov1[4,5]). In practice, because of the dimeric nature of M^{pro}, another strategy can be employed to inhibit its activity through the development of dimerization inhibitors.[2,6] Indeed, dimerization inhibitor design was previously reported for many viral enzymes such as the HIV reverse transcriptase, integrase, herpes simplex virus ribonucleotide reductase, and DNA polymerase.[6,7] In fact, targeting dimerization could potentially affect the substrate pocket and thus inhibit the M^{pro} activity because of allosteric connectivity between the dimerization site and the catalytic site.[2,8] Recently, we provided extensive simulations on M^{pro}[9] using the AMOEBA polarizable force field (PFF)[10−12] and a new highly parallel GPUs-accelerated[13,14] unsupervised adaptive sampling strategy.[9] These multimicrosecond simulations and their associated conformational spaces were compared to available non-PFF long-time scale simulation data from D. E. Shaw Research (DESRES)[15] and RIKEN Center for Biosystems Dynamics Research.[16] It was found[9] that AMOEBA results were closely correlated with experimental data, highlighting the observed strong flexibility of M^{pro}.[17] However, important differences in structural dynamics were observed compared to non-PFFs in key areas of the protease. For example, the overall richer conformational space led to enhanced volume cavities and to different solvation patterns within the active site. In order to drive further our high-resolution M^{pro} analysis, we present here a study of the factors structuring the dimerization interface as a function of different pH and solvation patterns. We particularly focus on the study of the role of many-body effects in the modeling of interfacial water and on their impact in allosteric interactions of the dimerization interface with other cavities/sites. To do so, we analyze more than 50 μs (including more than 12 μs of new simulations produced for the study) of AMOEBA molecular dynamics simulations and more than 110 μs of additional non-PFF simulations from other available data sets. All simulation details can be found in Theoretical Methods at the end of this Letter.

Received: May 6, 2021
Accepted: June 10, 2021
Published: July 1, 2021

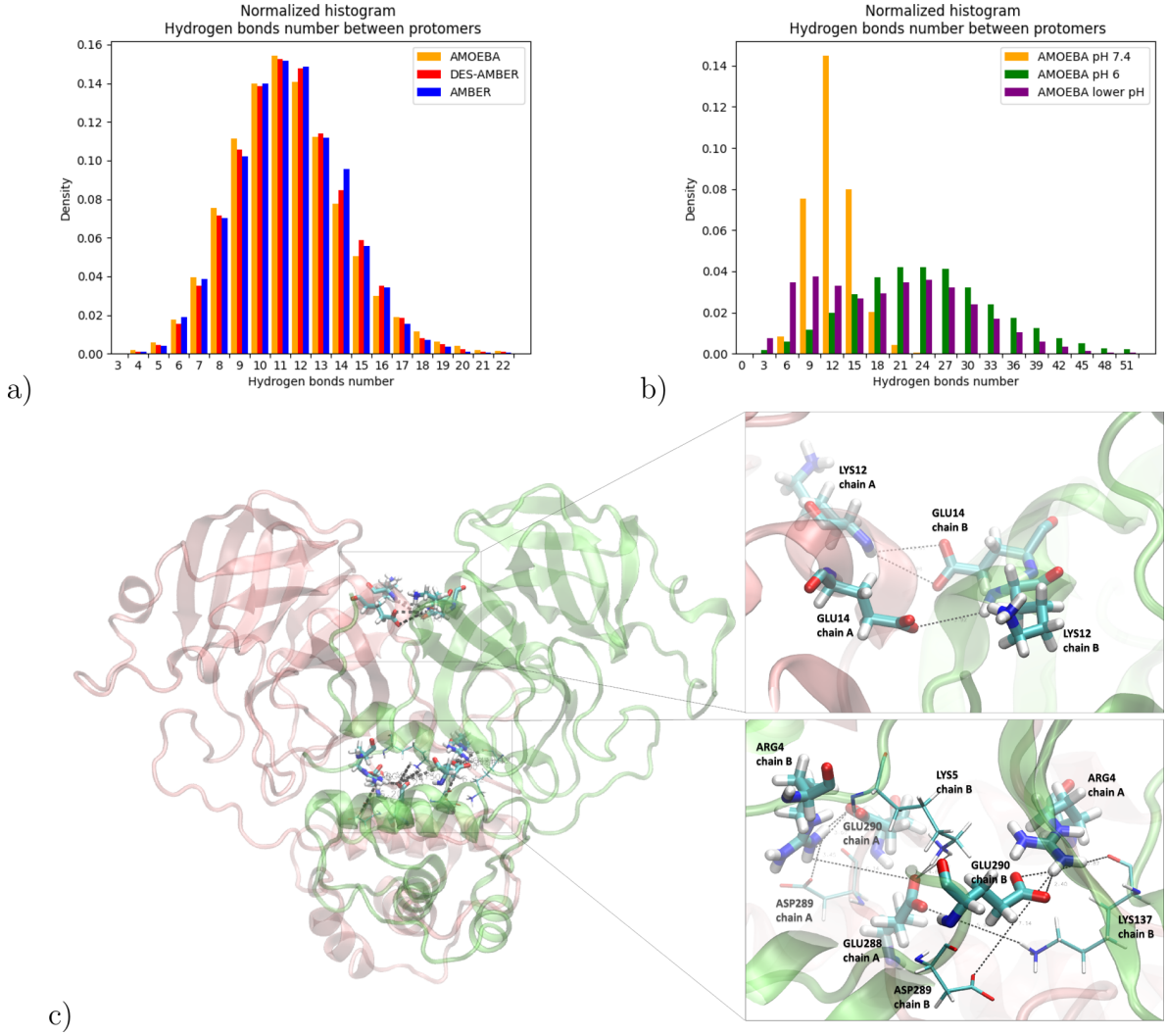


Figure 1. Histogram representation of H-bond probability density for (a) DES-AMBER, AMBER, and AMOEBA force fields at pH 7.4 and for AMOEBA trajectories at pH 7.4, 6, and lower. (b) Representation of the most frequent H-Bond interactions at the dimerization interface. Chains **A** and **B** are presented in pink and lime, respectively, (c).

To start our analysis of the M^{pro} structural dynamics at the dimerization interface, we determined the number of hydrogen bond (H-bond) interactions in order to evaluate the robustness of noncovalent interactions between the two protomers. Starting at physiological pH, we analyzed the DES-AMBER (DESRES), AMBER (RIKEN), and AMOEBA (Tinker-HP) trajectories (see Theoretical Methods for details) provided within the available conformation ensembles. We found relatively similar H-bond interaction probability density functions between the three profiles (see Figure 1a) that all present strong stability of the dimerization interface. Comparing the physiological H-bond distribution to lower pH AMOEBA simulations (see Figure 1b), we found a transition from a sharp Gaussian distribution centered at 14 H-bonds (pH 7.4) to a more diffuse one at pH 6 and below, exhibiting the involvements of weaker, disorganized, interactions. Clearly, our results show a collapse of the dimer interface at pH values lower than physiological as a consequence of the successive protonations of histidine residues (**His172** then **His163**).[9,18,19] Among the observed interactions (see Table 1 in the Supporting Information), **Arg4−Glu290** and **Gly11−Glu14** H-bond interactions have the highest probability density of all over DES-AMBER, AMBER, and AMOEBA trajectories at physiological pH. However, these interactions are not detected at lower pH, which is consistent with experimental studies reporting that low pH is responsible for the loss of the dimer interface.[20,21] It is important to note here that protonation of **His172** at lower pH has recently been shown[9,17,19] to be the source of a partial collapse in the catalytic site as well. Because the dimer interface is known to be fully functional at physiological pH, our multi-pH results reinforce the critical role of the **His172** protonation state and are consistent with Verma et al. findings[19] of a nonprotonated **His172** at physiological pH. A detailed look at the H-bond interaction profile in Table 1 of the Supporting Information highlights the key role of **Arg4** in maintaining the dimerization through several interactions, mainly with **Glu290** but also with **Lys137**, **Ser139**, **Glu288**, and **Asp289** at physiological pH. This is consistent with the description of key residues for the maintenance of SARS-CoV-2 M^{pro} dimerization in the experimental literature:[22] **Arg4**, **Ser10**, **Gly11**, **Glu14**, **Asn28**, **Ser139**, **Phe140**, **Ser147**, **Glu166**, **Glu290**, and **Arg298**. These residues all appear along our analysis, except for **Ser147**. Nevertheless, we were capable here of expanding

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

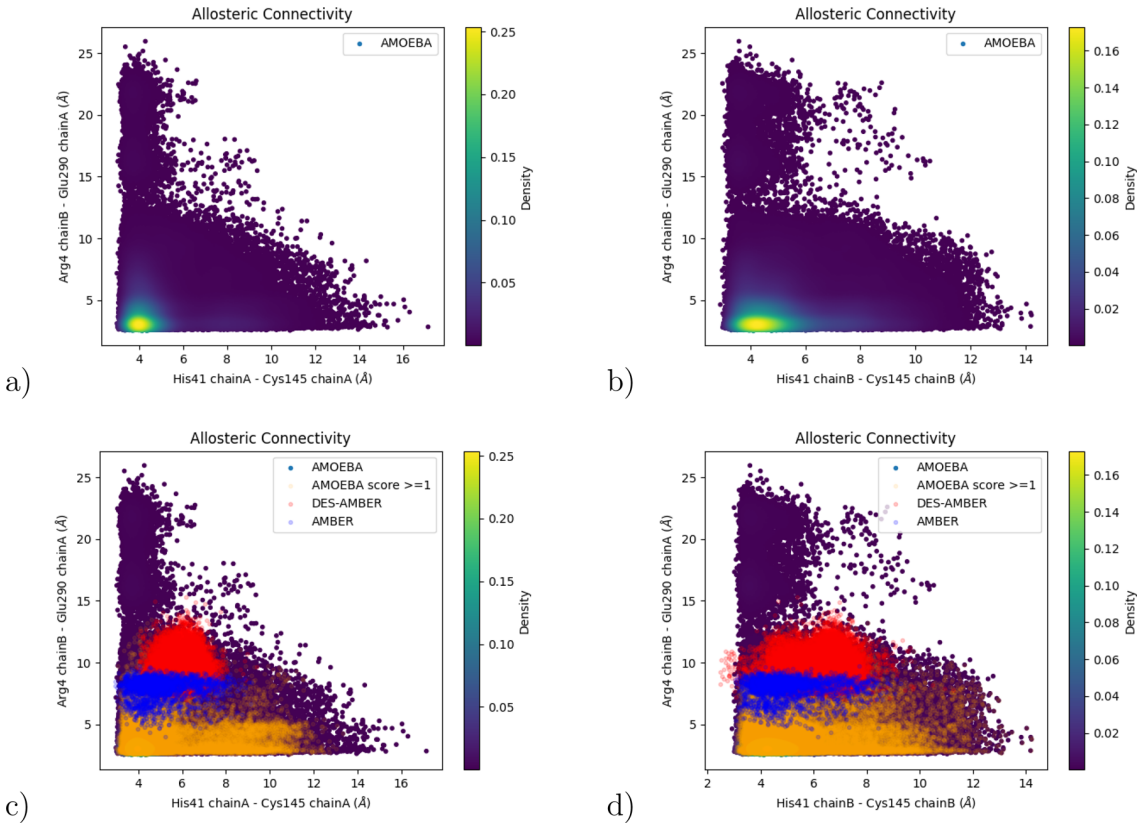


Figure 2. 2D plot representation of ***Arg4*** *chain* ***B−Glu290*** *chain* ***A*** distances vs ***His41*** *chain* ***A−Cys145*** *chain* ***A*** distances (a and c) and vs (b and d) His41 chain **B−Cys145** chain **B**. In panels c and d we have projected on the AMOEBA 15.14 μs, DES-AMBER 100 μs, AMBER 10 μs, and AMOEBA frames with a reweighting score greater than 1.

the list of these residues after a detailed analysis of DES-AMBER, AMBER, and AMOEBA simulations. As shown in Table 1 (Supporting Information), AMOEBA predicts a richer, more exhaustive, list of dimerization-implied residues compared to AMBER and DES-AMBER. The detected special forms of H-bond and other interactions, at physiological pH, are highlighted in Figure 1c. It is important to note that when successive histidine protonations occur, **His172** and **His163** switch from neutral histidines at pH 7.4 to positively charged at pH 6 and below, changing the nature of some of their interactions with other residues and water (for example, moving from H-bonds to salt-bridges in some cases[9,23]). Although pH lowering will affect also other residues that are not all considered in our computations,[19] this physicochemical change in the nature of the histidines interactions is central to the weakening of the interface stability, forcing it to redistribute its H-bond network into a different and less structured configuration. Finally, Table 1 (Supporting Information) also reveals that the Arg4−Glu290 and Gly11−Glu14 interactions are the most important H-bonds responsible for the stabilization of the dimerization interface because they exhibit the highest densities at physiological pH and are absent in the lower pH simulations. Overall, these results highlight the fact that the complex H-bond network is the one driving force stabilizing the interface.

To probe deeper into the complexity of the dimerization interface, we decided to look at its potential allosteric interactions within M^{pro}. Allostery occurs when conformational changes happening at one site of a protein and causing structural or dynamical changes at a topologically independent distant site. Such changes lead to a reduction or an increase in catalytic activity among other structural rearrangements. Structure-based prediction of allosteric sites, modulators, and communication pathway is important for a basic understanding of proteins and can lead drug discovery in order to regulate protein function.[24,25] Because H-bonds play a very important role in the dimerization region, they may be able to influence its volume, which could also have structural effects on other protein surface pockets via allosteric correlations.[24] The druggability of the dimerization interface has been discussed in the literature,[9,20] but fewer contributions looked at the potential allosteric interactions. Indeed, the importance of allosteric connectivity between allosteric and functional sites has been increasingly witnessed during recent years.[26,27] Several potential allosteric sites were recently discussed in order to offer allosteric drug target strategies[28−30] inside SARS-CoV-2 M^{pro}. For example, Stromich et al.[29] studied the scoring of putative allosteric sites and underlined a zone located in the dimerization site showing a high connectivity toward the catalytic active site. They proposed the definition of a potential allosteric dimerization site formed by the six following residues of the interface: **Arg131**, **Asp197**, **Thr199**, **Asp289**, and **Glu290** from chain **A** and **Arg4** from chain **B**. Because several of these residues were shown by our simulations to be instrumental to the interface stabilization (see Table 1, Supporting Information and previous discussion), we decided to study this site. In order to assess for a potential allosteric connectivity of the allosteric dimerization site toward both chains of the catalytic active site and to analyze its structural dynamics, we resorted to extensive bond-to-bond propensity

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

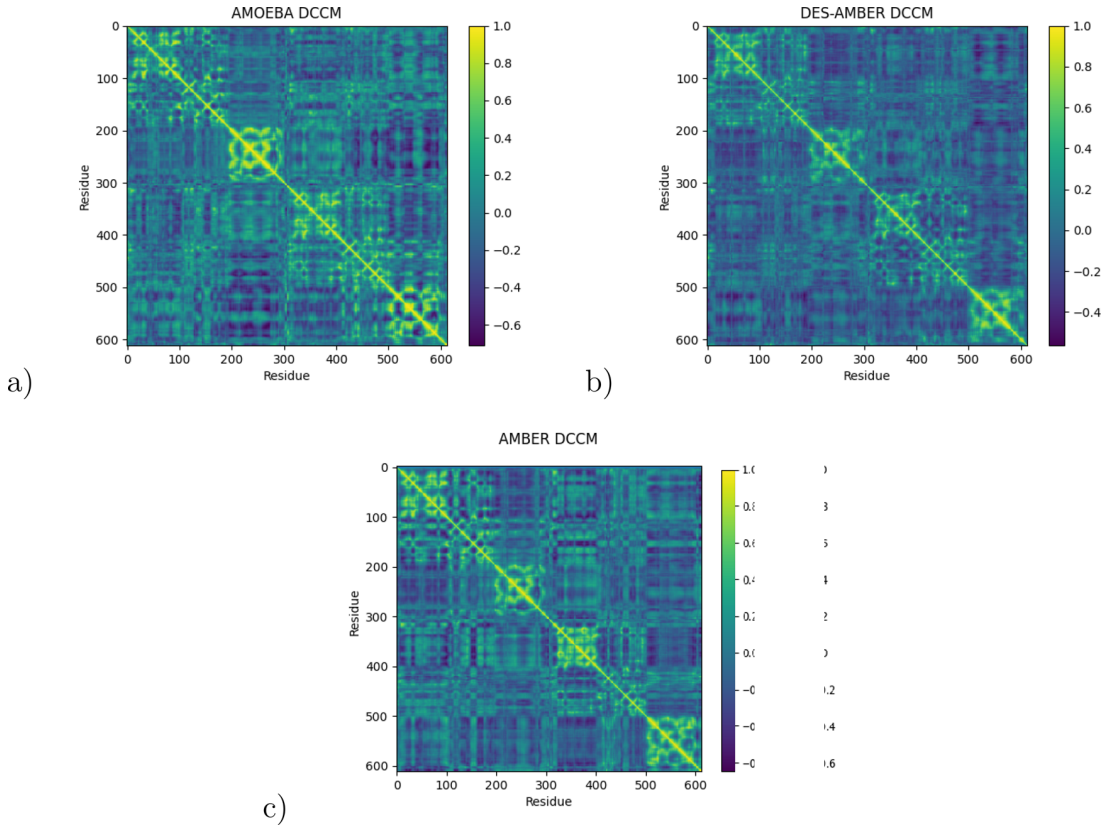


Figure 3. Dynamic cross-correlation maps using the C_α atom of each residue for (a) AMOEBA, (b) DES-AMBER, and (c) AMBER trajectories.

analysis.[31] Using this approach, we measure the fluctuations of given sets of atom−atom interactions and analyze how they affect any other set of interactions located elsewhere within the protein, allowing therefore to measure their instantaneous connectivity at each moment of the dynamics. We calculated first the evolution of distances located inside the allosteric dimerization site with other characteristic distances implicated in the residues forming the catalytic dyad. That way, thanks to well-chosen reference atoms or residues, this study informs us indirectly of the coevolution of the two cavity volumes. Indeed, comparing their volume fluctuations along trajectories can tell us about a possible allosteric connectivity between them.[29,32] We show in Figure 2a,b a 2D plot graphic of the distances separating the residues of the catalytic dyad for both chains **A** and **B** versus the distances between residues from the allosteric dimerization site: **Arg4** chain **B** and **Glu290** chain **A** because they present a robust interaction. AMOEBA trajectories show a high density of structures having both narrow catalytic and allosteric dimerization sites, respectively, around 4 and 3 Å, as shown in Figure 2a,b. However, we are also able to detect a different organization of the structures that are characterized by a narrow allosteric dimerization site and a relaxed catalytic site and, conversely, proposing possible allosteric connectivity between the sizes of the catalytic and allosteric dimerization sites. This additional connectivity found in the AMOEBA simulations is not observed in DES-AMBER nor in AMBER simulations (Figure 2c,d). Within our adaptive sampling scheme, the score is defined as the ratio between the probabilities to obtain the structure q_i in the biased simulation and in an unbiased simulation. Here, we limit ourselves to structures with a reweighting score greater than 1 as they are more likely to be visited during a conventional MD simulation. In contrast, frames with scores less than 1 have been favored by the adaptive algorithm to maximize exploration and are thus less physically relevant to the system statistic (more information can be found in ref 9). Thus, structures presented in orange in Figure 2 are more representative of the true AMOEBA statistics. In this case, we detect mostly structures having a relaxed catalytic site and a narrow allosteric dimerization site. This suggests that this specific dependency is detected thanks to the use of the polarizable AMOEBA FF, whereas the adaptive algorithm sampling is the one responsible for detecting structures associated with both a narrow catalytic site and a relaxed allosteric dimerization site. Similar conclusions can be reached upon considering **Arg131**, **Asp197**, and **Thr199** instead of **Glu290**, as shown in Figure 1 of the Supporting Information. These observations demonstrate the importance of the coupling of the adaptive sampling algorithm to the AMOEBA PFF for bringing out conformations that have escaped nonpolarizable standard MD simulations.

Because some allosteric connection was found between the dimerization and the active sites, we decided to provide another view of the simulation differences observed with the different force fields. To do so, we performed dynamic cross-correlation map (DCCM) analysis[33,34] for the three trajectories. DCCM allows us to investigate the dynamical changes of the system over time and to quantify the correlation coefficients of motions between atoms. The first result to point out is that as seen previously, AMOEBA data differ from the AMBER/DES-AMBER data. DCCM shows more positive/negative values than those obtained from non-PFFs, indicating a stronger correlated/anticorrelated atom motion in PFF simulations (see Figure 3). It is worth mentioning that strong

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

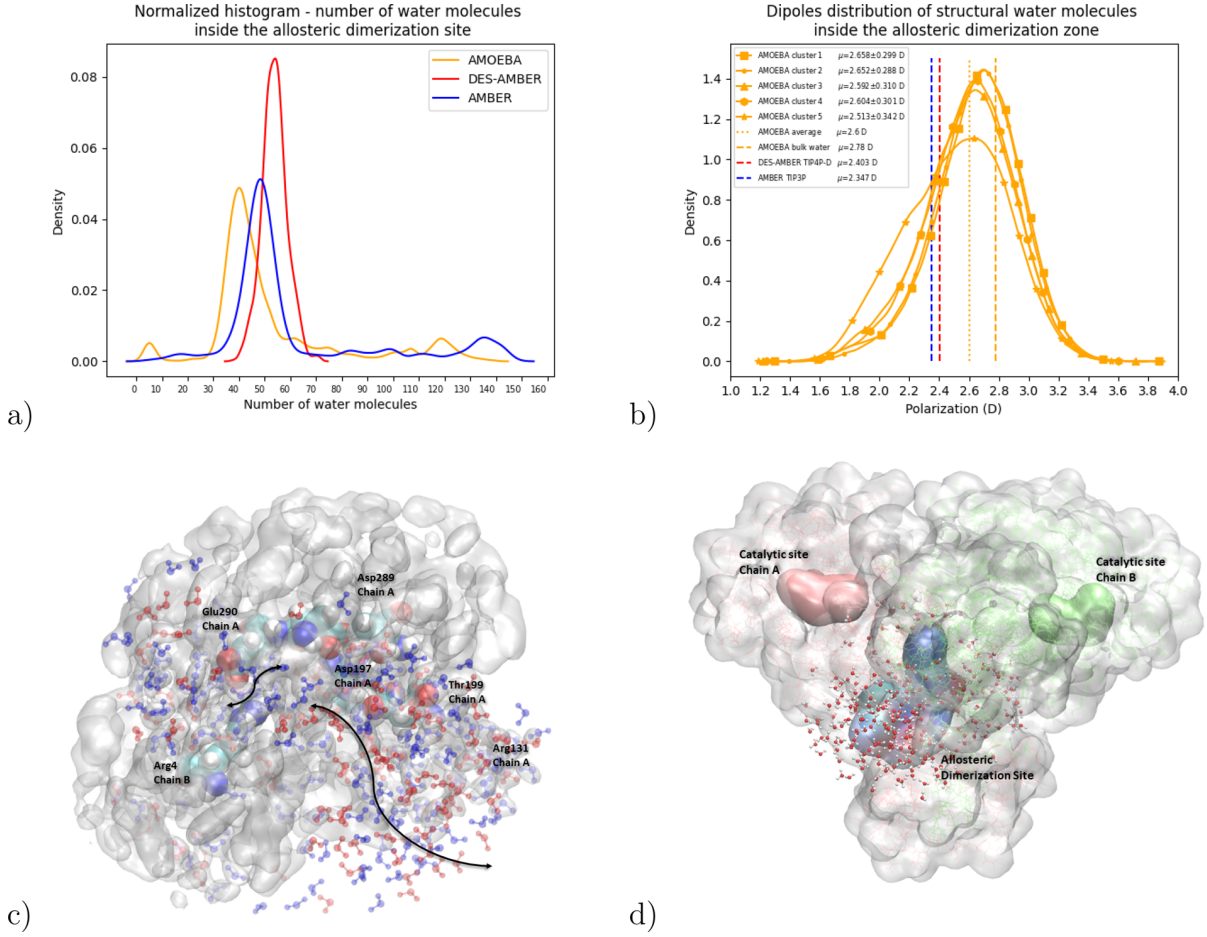


Figure 4. Representation of (a) the probability of structural water molecules number inside the allosteric dimerization site and (b) their dipoles distribution. (c) Representation of the water dipole distribution inside the allosteric dimerization site. Water molecules layered with red have dipole moment ≤2.78 D; those layered with blue have dipole moment ≥2.78 D. Asp and Glu have electrically charged side chains (acidic). Arg have electrically charged side chains (basic). Thr has polar side chain. The distance between **Arg4** and **Glu290** is 5.29 Å. Residues within 10 Å of the allosteric dimerization site are presented in quicksurf mode in white. Black arrows show the flow of water molecules in this site. (d) Global view of the M^{pro}, showing the catalytic site of both chain **A** and **B** and the allosteric dimerization site. Water molecules within 10 Å of the allosteric dimerization site are presented in cpk mode.

anticorrelation motions are observed between the α-helical region of each protomer of M^{pro} (a region strongly participating to the dimerization, i.e., residue range of 220−280 and 470−570) in AMOEBA trajectories. By contrast, the corresponding regions have much weaker (anti)correlation in both DES-AMBER and AMBER trajectories. Figure 2 in the Supporting Information proposes a closer analysis of the regions of interest for the allosteric interactions (i.e., the allosteric dimerization site) and reveals a more global anticorrelated motion between the residues of the allosteric dimerization site and the catalytic dyad of chain **A** than in AMBER/DES-AMBER. For chain **B**, this anti-correlation of the dimerization site with the catalytic dyad residues is also found. In all cases, the stronger correlation DDCM values are found within the AMOEBA simulation. The most positive correlation is found for Cys145 (chain B) and Arg4 (chain B) as the most negative correlation is found for Cy145 (chain A) and Glu290 (chain A). This further confirms the presence of an allosteric correlation between the sites and also supports the hypothesis of a strong asymmetry between protomers.[9]

As our previous analysis confirmed the differences between FF simulations, resulting in different predictions of allosteric connections and correlated motions between sites, we attempted to trace back the discrepancies studying the overall structural dynamics of the interface. As we explained in the first section, the dimerization interface overall stability is linked to a complex H-bond network that is exposed to the water solvent. Within M^{pro}, cavities and pocket volume fluctuations lead to water molecule traffic which is essential to maintain the protein structure. In a sense, the allosteric connection is performed "through water" and the resulting analysis of its presence is therefore impacted by the quality of water modeling. In practice, water molecules are commonly found within enzymatic sites, can form water bridges between the residues, and thus maintain protein secondary structures via H-bond interactions (see ref 35 and references therein). Using polarizable force fields, it has been demonstrated that some structural water molecules exhibit enhanced dipole moments, in kinase active sites for example.[36] Our previous work on M^{pro} clearly also demonstrated a very different behavior of water molecules when they are modeled with the AMOEBA PFF, which takes into account many-body effects.[9] Because water plays an important role in structural and functional activities, we looked for the water molecules present around some key

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

interface residues at physiological pH. To do so, we considered a 3.5 Å radius sphere centered at the atom capable of being engaged in hydrogen bonds with water for the most important residues involved in noncovalent interactions between protomers, namely: **Arg4**, **Glu290**, **Gly11**, and **Glu14**. The number of detected water molecules (see Figure 3 in the Supporting Information), presents notably different distribution profiles depending on the simulations: AMOEBA polarizable water, DES-AMBER(TIP4D), and AMBER (TIP3P). In fact, the number of water molecules detected strongly depends on the type of residue, on the considered M^{pro} chain, and on the force field itself. **Arg4** of chain **A**, for example, is found to be mostly interacting with one water molecule for AMBER, 1−2 molecules for DES-AMBER, and 2−3 molecules for AMOEBA. However, **Arg4** of chain **B** is found to interact mostly with 3 water molecules for AMBER and DES-AMBER and with 2 molecules for AMOEBA in line with the predicted asymmetry between protomers found in M^{pro}.[9] Although water traffic is detected for all force fields, the solvation patterns and differences between force fields appear to be residue-dependent. Water molecules extracted from AMOEBA trajectories around the concerned residues are polarizable (and the water model is flexible[10]), and therefore, their distribution is mainly controlled by the physicochemical nature of the residues (polar, apolar, positively/negatively charged, etc.) generating specific polarizing fields. In practice, the AMOEBA bulk water average dipole moment amounts to 2.78 D, in nice agreement with experiment, whereas non-PFF models exhibit smaller fixed dipole moments of 2.40 and 2.35 D for TIP4P-D and TIP3P, respectively. Figure 4 in the Supporting Information shows the average dipole values for the water molecules in the vicinity of the targeted residues. Their mean values (around 2.6 D on average) is below the bulk AMOEBA reference value. This result is consistent with the idea that the dense interface environment generates a global many-body depolarizing effect (compared to bulk water) influencing the water molecule-induced dipoles. Overall, the interface H-bond network connects to the solvent's own H-bond pattern forming a higher level of complexity. Clearly, the water molecule behavior is strongly influenced by the nature of the interface residues through many-body effects, generating various microsolvation patterns according to the local environment. These patterns are themselves affected by their interactions with the solvent in a self-consistent fashion.

In order to further evaluate the difference in solvation patterns, we focused on the previously introduced allosteric dimerization site, a specific location within the interface that allows for water molecules to circulate between the interface residues. To get a better understanding of what is happening, we have to evaluate the number of water molecules present and their lifetimes within this site. It is important to mention here that the six residues forming the allosteric site at the dimerization interface are either ionic or polar. Asp and Glu are negatively charged, whereas His is positively charged. Side-chains such as Thr can retain water molecules inside the cavity. Black arrows in Figure 4 display the flow of water molecules in the buried site. Because the greatest distance separating **Arg4** chain **B** and **Glu290** chain **A** is around 24 Å, we defined a sphere with a (cutoff) radius of 10 Å, centered at the geometrical center of the six residues forming the pocket at the allosteric dimerization site, and calculated the number of water molecules present within this sphere. Figure 4a shows a striking difference between AMOEBA and non-PFF simulations. PFF simulations give far fewer water molecules inside the allosteric dimerization site and a highest probability density of presence centered at 40, to be compared with 50 for AMBER and 55 for DES-AMBER.

We then measured the water lifetimes in the 10 Å sphere using the 400 ns CMD simulations produced with both the AMBER and AMOEBA force fields. We observed an average water lifetime of 0.171 ns for AMBER and a longer lifetime of 0.516 ns for AMOEBA. This clearly shows that many-body polarization effects tend to act as glue between the dimerization interface and the water molecules, specifically at the allosteric dimerization site, retaining them longer at the surface of the residues of the dimerization site (Figure 5 in the Supporting Information). Putting these two findings together allows us to better understand why the water dynamics outside the interface is so different from the (slower) dynamics found in the most confined part of the dimerization allosteric site. The smaller number of water molecules inside the allosteric dimerization site reflects therefore a slower water traffic, because these polarized water molecules tend to move slowly, being engaged into many more H-bonds. Indeed, the AMOEBA diffusion constant is more in line with experiment than the TIP3P and TIP4-D models. However, as we discussed, the AMOEBA water dipole moment values can present strong local variations because of the local microsolvation patterns that cannot be captured by the mean-field approximation, which is the basis of classical non-PFFs.[35] As for the previous situation, Figure 4 displays a rather underpolarized global situation for water that exhibits an average dipole moment lower than that of the bulk. Nevertheless, Figure 4 also highlights the collection of multiple different situations where the microsolvation patterns tend to generate simultaneously partial distributions of highly polarized and underpolarized water molecules in the allosteric dimerization site because this distribution is mainly controlled by the physicochemical nature of the residues. As shown in Figure 4c and in Figure 6 in the Supporting Information, mostly underpolarized water molecules are found in the most buried section of the allosteric dimerization site where confinement generates more depolarizing effects. These are well-known to decrease the average dipole moment values of confined waters and are observed here. Again, AMOEBA exhibits a higher probability density lower than bulk at 2.6 D, whereas DES-AMBER and AMBER water dipoles remain fixed at 2.403 and 2.347 D, respectively (see Figure 4b). Figures 4b also provides a view of the average dipole moments found after clusterization of the AMOEBA trajectories (see ref 9 for more information about the five different clusters). The site maintains a relatively stable average dipole solvent value because of the fluctuation of both the volumes (i.e., different in the different clusters) and the number of water molecules (see Figure 7 in the Supporting Information), highlighting the interconnection of the interface H-bond network and the solvent. This suggests that there is a complex interplay between the distribution of dipoles of polarizable water molecules and the residues (and associated volumes) of the dimerization allosteric site. This interaction network contributes to regulating the allosteric effects with the catalytic site of both protomers. Modeling such connections between cavities requires capturing the subtle equilibrium between the protein and solvent dynamics. The dipolar fluctuations of the water traffic tend to be extremely complex, leading to dramatically different behavior in different parts of the interface where the

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

local water dynamics can be quite different (i.e., for the AMOEBA-predicted dynamic slowdown within the buried allosteric dimerization site, etc.). Such water traffic shapes the interface and participates in modulating the allosteric dimerization site structural "breathing" that is involved in the overall allosteric effects with the main catalytic site. Such critical involvement of the "polarizable" water molecule within recognition or regulatory sites of proteins had been postulated before,[36] and it is clear that the number of water molecules within a binding site matters. Indeed, waters interacting with their close environment via through-water binding modes are common and able to strongly influence local electronic properties.[37] Through-water configurations can mediate interactions between an inhibitor (see for example refs 36 and 38) and indirectly bound residues of the recognition site. In such situations, also considered in the context of pFFs, an accurate count of water molecules can be critical because many-body effects (particularly the polarization energy) could tip the (free) energy balance between competing inhibitors. Missing this aspect within the modeling certainly results in a loss in the prediction of signal in the allosteric communication. It is also important to mention that beyond this energetic view of the phenomenon, the connection between interfacial water molecules and protein dynamics/flexibility has been extensively discussed in the experimental literature (see references 39−41 and references therein): protein dynamics and solvation shell dynamics have been characterized regionally. More precisely, it has been observed that flexible regions of proteins generally encompass fast-moving waters, while stable regions are embedded into slower hydration layer water molecules. This is exactly what we see here, and what is new in our results is that such regional dynamics modeling is shown to be strongly affected by many-body effects. Indeed, they strongly influence the dynamics of interfacial water molecules acting on their local "viscosity" and therefore local dynamics. As binding pockets and allosteric sites require being reasonably stable over time to be targeted by drugs, in some situations, non-PFF simulations may tend to predict solvation patterns associated with an excessive water traffic and to too fast-moving interfacial molecules. This could unfortunately lead to the destabilization of druggable hotspots that therefore would potentially remain unknown to molecular modelers.

To conclude, in order to propose a high-quality model of the dimerization interface of SARS-CoV2 M^{pro} that could be used for further drug design, it is important to understand well and model its complex H-bonds network that is embedded within a dynamic dipolar water solvent network. Water appears to be a key player in the overall structural dynamics of the dimerization interface, being one building block of the global allosteric effects between sites through many-body polarization interactions with the interface residues. As we stressed before,[9] M^{pro} is a difficult and complex molecular system that requires the simultaneous ability to (i) accurately describe all types of noncovalent interactions within the protein and solvent requiring therefore an accurate force field able to describe local many-body polarization effects and (ii) perform extensive sampling going beyond the microsecond time scale. Of course, we analyzed here only one example of allosteric interactions within M^{pro} and many other ones may remain to be discovered; we hope that these analyses and molecular dynamics trajectories (available via the BioExcel/MolSSI repository) will help drug hunters targeting the M^{pro} dimerization interface.

THEORETICAL METHODS

To study the dimerization interface we extensively analyzed the all-atom conformation space produced previously[9] using the AMOEBA polarizable force field (AMOEBA protein force field[11,12] and AMOEBA03 flexible water model[10]) as well as the one provided by the RIKEN[16] (using the AMBER ff14SB force field[42] and the TIP3P water model[43]) and DESRES[15] (using the DES-AMBER[44] and TIP4P-D water model[45]) groups. Following the same simulation protocol (reference PDB structure 6LU7[46]) proposed in our previous work,[9] we performed separate additional runs of adaptive simulations for a total of 12 μs with AMOEBA to simulate low pH values. In this case, additional histidine residue protonation occurs. Therefore, to produce additional data to the pH 7.4 and pH 6 simulations proposed in our previous data set,[9] we also successively protonated (2 × 6 μs runs) the two **His163** residues to simulate further pH lowering (see discussion and Table 2 in ref 18). Further 800 ns AMOEBA and AMBER99SB conventional molecular dynamics simulations (400 ns × 2) were produced at physiological pH and restarting from starting points from our previous data set, taking a snapshot every 10 ps to enable an in-depth analysis of the role of the water solvent. All additional all-atom simulations were performed using the newly developed GPUs module[14] within the Tinker−HP package,[13] which is part of the Tinker 8 platform.[47] This recently developed module is able to efficiently leverage mixed precision,[14] offering a strong acceleration of simulations using GPUs. Periodic boundary conditions using a cubic box of side length 100 Å were used. Langevin molecular dynamics simulations were performed using the BAOAB−RESPA1 integrator[48] using a 10 fs outer time step, a preconditioned conjugate gradient polarization solver (with a 10^{-5} convergence threshold), hydrogen−mass repartitioning (HMR), and random initial velocities. Periodic boundary conditions (PBC) were employed using the smooth particle mesh Ewald (SPME) method with a grid of dimension 128 Å × 128 Å × 128 Å. The Ewald-cutoff was taken to 7 Å, and the van der Waals cutoff was taken to be 9 Å. Post processing analysis was done using the MDTraj,[49] Scikit-Learn,[50] and Scipy packages.[51] Dynamical cross-correlation matrices (DCCMs) were generated based on the C_α atom of each residue by using the functionality provided in the MD-TASK package.[52]

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpclett.1c01460.

All the residues implicated in H-bond interactions (Table 1); 2D plot representation of distances **His41**$_X$**−Cys145**$_X$ (X = chain **A** or chain **B**) versus distances **Arg4**$_B$**−Arg131**$_A$, **Arg4**$_B$**−Asp197**$_A$ and **Arg4**$_B$**−Thr199**$_A$ showing that allosteric connectivity is present (Figure 1); extracted values from dynamic cross-correlation maps revealing the cross-correlation between residues implicated in allosteric connectivity (Figure 2); number of water molecules detected in a 3.5 Å radius from **Arg4**$_X$, **Gly11**$_X$, **Glu14**$_X$, or **Glu290**$_X$ (X = chain **A** or chain **B**) (Figure 3); dipole distribution of structural water molecules interacting with **Arg4**$_X$, **Gly11**$_X$, **Glu14**$_X$, or **Glu290**$_X$ (X = chain **A** or chain **B**) (Figure 4); water lifetime distribution inside the allosteric

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

dimerization site (Figure 5); representation of the water dipole distribution inside the allosteric dimerization site, for 5.29 and 8.7 Å between **Arg4** and **Glu290** (Figure 6); 2D plot representation of the volume of the dimerization site vs the number of water molecules inside the allosteric dimerization site and schematic representation of the SARS-CoV-2 M^{pro} dimer showing the dimerization site and the allosteric dimerization site residues (Figure 7) (PDF)

AUTHOR INFORMATION

Corresponding Author

Jean-Philip Piquemal − *Sorbonne Université, 75005 Paris, France; Institut Universitaire de France, 75005 Paris, France; Department of Biomedical Engineering, University of Texas at Austin, Austin, Texas 78712, United States;* orcid.org/0000-0001-6615-9426; Email: jean-philip.piquemal@sorbonne-universite.fr

Authors

Dina El Ahdab − *Sorbonne Université, 75005 Paris, France; Université Saint-Joseph de Beyrouth, 1104 2020 Beirut, Lebanon*

Louis Lagardère − *Sorbonne Université, 75005 Paris, France; Sorbonne Université, 75005 Paris, France*

Théo Jaffrelot Inizan − *Sorbonne Université, 75005 Paris, France*

Fréderic Célerse − *Sorbonne Université, 75005 Paris, France; Sorbonne Université, 75005 Paris, France;* Present Address: F.C.: EPFL, LCMD, Lausanne, 1015, Switzerland.; orcid.org/0000-0001-8584-6547

Chengwen Liu − *Department of Biomedical Engineering, University of Texas at Austin, Austin, Texas 78712, United States;* orcid.org/0000-0002-3930-7793

Olivier Adjoua − *Sorbonne Université, 75005 Paris, France*

Luc-Henri Jolly − *Sorbonne Université, 75005 Paris, France*

Nohad Gresh − *Sorbonne Université, 75005 Paris, France;* orcid.org/0000-0001-7174-2907

Zeina Hobaika − *Université Saint-Joseph de Beyrouth, 1104 2020 Beirut, Lebanon*

Pengyu Ren − *Department of Biomedical Engineering, University of Texas at Austin, Austin, Texas 78712, United States;* orcid.org/0000-0002-5613-1910

Richard G. Maroun − *Université Saint-Joseph de Beyrouth, 1104 2020 Beirut, Lebanon*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jpclett.1c01460

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No 810367), project EMC2 (J.-P.P.). D.E.A. acknowledges funding from the Lebanese National Council for Scientific Research, CNRS-L. F.C. acknowledges funding from the French state funds managed by the CalSimLab LABEX and the ANR within the Investissements d'Avenir program (reference ANR11-IDEX-0004-02) and support from the Direction Génerale de l'Armement (DGA) Maîtrise NRBC of the French Ministry of Defense. Adaptive sampling computations have been performed at GENCI thanks to a COVID19 emergency allocation on the Jean Zay machine (IDRIS, Orsay, France) on Grant No. A0070707671 and on the Irene Joliot Curie machine thanks to a PRACE COVID-19 emergency grant (Project COVID-HP). Additional conventional AMOEBA and AMBER MD simulations have been performed on the Amazon cloud platform thanks to an AWS COVID-19 special grant. The authors thank the Swiss National Supercomputing Center (CSCS) for hosting our data through the FENIX infrastructure. J.-P.P. acknowledges a special COVID-19 funding from Sorbonne Université. P.R. is grateful for support by National Science Foundation (CHE-1856173) and National Institutes of Health (R01GM106137 and R01GM114237).

REFERENCES

(1) Zumla, A.; Chan, J. F.; Azhar, E. I.; Hui, D. S.; Yuen, K.-Y. Coronaviruses-Drug Discovery and Therapeutic Options. *Nat. Rev. Drug Discovery* **2016**, *15*, 327−347.

(2) Ding, L.; Zhang, X.-X.; Wei, P.; Fan, K.; Lai, L. The interaction between severe acute respiratory syndrome coronavirus 3C-like proteinase and a dimeric inhibitor by capillary electrophoresis. *Anal. Biochem.* **2005**, *343*, 159−165.

(3) Cui, W.; Yang, K.; Yang, H. Recent Progress in the Drug Development Targeting SARS-CoV-2 Main Protease as Treatment for COVID-19. *Front. Biosci.* **2020**, *7*, 398.

(4) Pillaiyar, T.; Manickam, M.; Namasivayam, V.; Hayashi, Y.; Jung, S.-H. An Overview of Severe Acute Respiratory Syndrome−Coronavirus (SARS-CoV) 3CL Protease Inhibitors: Peptidomimetics and Small Molecule Chemotherapy. *J. Med. Chem.* **2016**, *59*, 6595−6628.

(5) Kuo, C.; Liang, P. Characterization and Inhibition of the Main Protease of Severe Acute Respiratory Syndrome Coronavirus. *ChemBioEng Rev.* **2015**, *2*, 118−132.

(6) Boggetto, N.; Reboud-Ravaux, M. Dimerization Inhibitors of HIV-1 Protease. *Biol. Chem.* **2002**, *383*, 1321−1324.

(7) Zutshi, R.; Brickner, M.; Chmielewski, J. Inhibiting the assembly of protein-protein interfaces. *Curr. Opin. Chem. Biol.* **1998**, *2*, 62−66.

(8) Wei, P.; Fan, K.; Chen, H.; Ma, L.; Huang, C.; Tan, L.; Xi, D.; Li, C.; Liu, Y.; Cao, A.; Lai, L. The N-terminal octapeptide acts as a dimerization inhibitor of SARS coronavirus 3C-like proteinase. *Biochem. Biophys. Res. Commun.* **2006**, *339*, 865−872.

(9) Jaffrelot Inizan, T.; Célerse, F.; Adjoua, O.; El Ahdab, D.; Jolly, L.-H.; Liu, C.; Ren, P.; Montes, M.; Lagarde, N.; Lagardère, L.; Monmarché, P.; Piquemal, J.-P. High-resolution mining of the SARS-CoV-2 main protease conformational space: supercomputer-driven unsupervised adaptive sampling. *Chem. Sci.* **2021**, *12*, 4889−4907.

(10) Ren, P. Y.; Ponder, J. W. Polarizable Atomic Multipole Water Model for Molecular Mechanics Simulation. *J. Phys. Chem. B* **2003**, *107*, 5933−5947.

(11) Shi, Y.; Xia, Z.; Zhang, J.; Best, R.; Wu, C.; Ponder, J. W.; Ren, P. Polarizable Atomic Multipole-Based AMOEBA Force Field for Proteins. *J. Chem. Theory Comput.* **2013**, *9*, 4046−4063.

(12) Zhang, C.; Lu, C.; Jing, Z.; Wu, C.; Piquemal, J.-P.; Ponder, J. W.; Ren, P. AMOEBA Polarizable Atomic Multipole Force Field for Nucleic Acids. *J. Chem. Theory Comput.* **2018**, *14*, 2084−2108.

(13) Lagardère, L.; Jolly, L.-H.; Lipparini, F.; Aviat, F.; Stamm, B.; Jing, Z. F.; Harger, M.; Torabifard, H.; Cisneros, G. A.; Schnieders, M. J.; Gresh, N.; Maday, Y.; Ren, P. Y.; Ponder, J. W.; Piquemal, J.-P. Tinker-HP: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields. *Chem. Sci.* **2018**, *9*, 956−972.

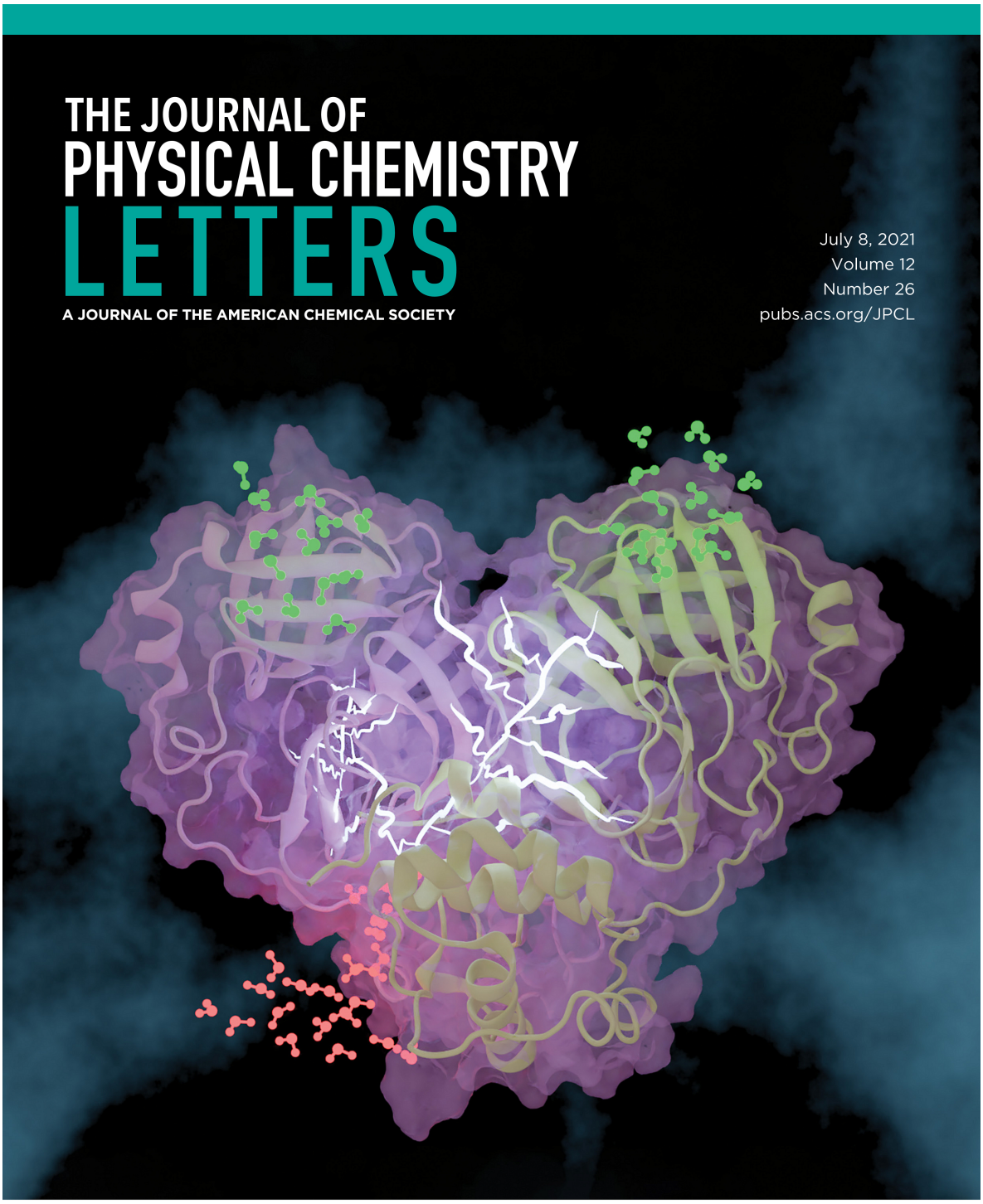
(14) Adjoua, O.; Lagardère, L.; Jolly, L.-H.; Durocher, A.; Very, T.; Dupays, I.; Wang, Z.; Inizan, T. J.; Célerse, F.; Ren, P.; Ponder, J. W.; Piquemal, J.-P. Tinker-HP: Accelerating Molecular Dynamics Simulations of Large Complex Systems with Advanced Point Dipole

Polarizable Force Fields Using GPUs and Multi-GPU Systems. *J. Chem. Theory Comput.* **2021**, *17*, 2034−2053.
(15) D. E. Shaw Research. *Molecular Dynamics Simulations Related to SARS-CoV-2,*. D. E. Shaw Research Technical Data, 2020, http://www.deshawresearch.com/resources_sarscov2.html.
(16) Komatsu, T. S.; Koyama, Y.; Okimoto, N.; Morimoto, G.; Ohno, Y.; Taiji, M. COVID-19 related trajectory data of 10 microseconds all atom molecular dynamics simulation of SARS-CoV-2 dimeric main protease, V2. *Mendeley Data* **2020**, *10*, 17632.
(17) Kneller, D. W.; Phillips, G.; O'Neill, H. M.; Jedrzejczak, R.; Stols, L.; Langan, P.; Joachimiak, A.; Coates, L.; Kovalevsky, A. Structural plasticity of SARS-CoV-2 3CL M pro active site cavity revealed by room temperature X-ray crystallography. *Nat. Commun.* **2020**, *11*, 3202.
(18) Tan, J.; Verschueren, K. H.; Anand, K.; Shen, J.; Yang, M.; Xu, Y.; Rao, Z.; Bigalke, J.; Heisen, B.; Mesters, J. R.; Chen, K.; Shen, X.; Jiang, H.; Hilgenfeld, R. pH-dependent Conformational Flexibility of the SARS-CoV Main Proteinase (Mpro) Dimer: Molecular Dynamics Simulations and Multiple X-ray Structure Analyses. *J. Mol. Biol.* **2005**, *354*, 25−40.
(19) Verma, N.; Henderson, J. A.; Shen, J. Proton-Coupled Conformational Activation of SARS Coronavirus Main Proteases and Opportunity for Designing Small-Molecule Broad-Spectrum Targeted Covalent Inhibitors. *J. Am. Chem. Soc.* **2020**, *142*, 21883−21890.
(20) Goyal, B.; Goyal, B. Targeting the Dimerization of the Main Protease of Coronaviruses: A Potential Broad-Spectrum Therapeutic Strategy. *ACS Comb. Sci.* **2020**, *22*, 297−305.
(21) Chou, C.-Y.; Chang, H.-C.; Hsu, W.-C.; Lin, T.-Z.; Lin, C.-H.; Chang, G.-G. Quaternary Structure of the Severe Acute Respiratory Syndrome (SARS) Coronavirus Main Protease. *Biochemistry* **2004**, *43*, 14958−14970.
(22) Liang, J.; Karagiannis, C.; Pitsillou, E.; Darmawan, K. K.; Ng, K.; Hung, A.; Karagiannis, T. C. Site mapping and small molecule blind docking reveal a possible target site on the SARS-CoV-2 main protease dimer interface. *Comput. Biol. Chem.* **2020**, *89*, 107372.
(23) Liao, S.-M.; Du, Q.-S.; Meng, J.-Z.; Pang, Z.-W.; Huang, R.-B. The multiple roles of histidine in protein interactions. *Chem. Cent. J.* **2013**, *7*, 44.
(24) Monod, J.; Wyman, J.; Changeux, J.-P. On the nature of allosteric transitions: a plausible model. *J. Mol. Biol.* **1965**, *12*, 88−118.
(25) Greener, J. G.; Sternberg, M. J. Structure-based prediction of protein allostery. *Curr. Opin. Struct. Biol.* **2018**, *50*, 1−8.
(26) Laskowski, R. A.; Gerick, F.; Thornton, J. M. The structural basis of allosteric regulation in proteins. *FEBS Lett.* **2009**, *583*, 1692.
(27) Suplatov, D. A.; Švedas, V. Study of Functional and Allosteric Sites in Protein Superfamilies. *Acta. Naturae* **2015**, *7*, 34−45.
(28) Günther, S.; et al. X-ray screening identifies active site and allosteric inhibitors of SARS-CoV-2 main protease. *Science* **2021**, *372*, 642−646.
(29) Strömich, L.; Wu, N.; Barahona, M.; Yaliraki, S. N. Allosteric Hotspots in the Main Protease of SARS-CoV-2. *BioRxiv* 2020, DOI: 10.1101/2020.11.06.369439.
(30) Carli, M.; Sormani, G.; Rodriguez, A.; Laio, A. Candidate Binding Sites for Allosteric Inhibition of the SARS-CoV-2 Main Protease from the Analysis of Large-Scale Molecular Dynamics Simulations. *J. Phys. Chem. Lett.* **2021**, *12*, 65−72.
(31) Amor, B. R. C.; Schaub, M. T.; Yaliraki, S. N.; Barahona, M. Prediction of allosteric sites and mediating interactions through bond-to-bond propensities. *Nat. Commun.* **2016**, *7*, 12477.
(32) La Sala, G.; Decherchi, S.; De Vivo, M.; Rocchia, W. Allosteric Communication Networks in Proteins Revealed through Pocket Crosstalk Analysis. *ACS Cent. Sci.* **2017**, *3*, 949−960.
(33) McCammon, J. A.; Harvey, S. C. *Dynamics of proteins and nucleic acids*; Cambridge University Press, 1988.
(34) Ichiye, T.; Karplus, M. Collective motions in proteins: a covariance analysis of atomic fluctuations in molecular dynamics and normal mode simulations. *Proteins: Struct., Funct., Genet.* **1991**, *11*, 205−217.
(35) Melcr, J.; Piquemal, J.-P. Accurate biomolecular simulations account for electronic polarization. *Front. Mol. Biosci.* **2019**, *6*, 143.
(36) de Courcy, B.; Piquemal, J.-P.; Garbay, C.; Gresh, N. Polarizable Water Molecules in Ligand-Macromolecule Recognition. Impact on the Relative Affinities of Competing Pyrrolopyrimidine Inhibitors for FAK Kinase. *J. Am. Chem. Soc.* **2010**, *132*, 3312−3320.
(37) de Courcy, B.; Pedersen, L. G.; Parisel, O.; Gresh, N.; Silvi, B.; Pilme, J.; Piquemal, J.-P. Understanding Selectivity of Hard and Soft Metal Cations within Biological Systems Using the Subvalence Concept. 1. Application to Blood Coagulation:Direct Cation−Protein Electronic Effects versus Indirect Interactions through Water Networks. *J. Chem. Theory Comput.* **2010**, *6*, 1048−1063.
(38) Gresh, N.; de Courcy, B.; Piquemal, J.-P.; Foret, J.; Courtiol-Legourd, S.; Salmon, L. Polarizable Water Networks in Ligand−Metalloprotein Recognition. Impact on the Relative Complexation Energies of Zn-Dependent Phosphomannose Isomerase with d-Mannose 6-Phosphate Surrogates. *J. Phys. Chem. B* **2011**, *115*, 8304−8316.
(39) Dahanayake, J. N.; Mitchell-Koch, K. R. How Does Solvation Layer Mobility Affect Protein Structural Dynamics? *Front. Biosci.* **2018**, *5*, 65.
(40) Bellissent-Funel, M.-C.; Hassanali, A.; Havenith, M.; Henchman, R.; Pohl, P.; Sterpone, F.; van der Spoel, D.; Xu, Y.; Garcia, A. E. Water Determines the Structure and Dynamics of Proteins. *Chem. Rev.* **2016**, *116*, 7673−7697.
(41) Combet, S.; Zanotti, J.-M. Further evidence that interfacial water is the main "driving force" of protein dynamics: a neutron scattering study on perdeuterated C-phycocyanin. *Phys. Chem. Chem. Phys.* **2012**, *14*, 4927−4934.
(42) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. *J. Chem. Theory Comput.* **2015**, *11*, 3696−3713.
(43) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926−935.
(44) Piana, S.; Robustelli, P.; Tan, D.; Chen, S.; Shaw, D. E. Development of a Force Field for the Simulation of Single-Chain Proteins and Protein−Protein Complexes. *J. Chem. Theory Comput.* **2020**, *16*, 2494−2507.
(45) Piana, S.; Donchev, A. G.; Robustelli, P.; Shaw, D. E. Water Dispersion Interactions Strongly Influence Simulated Structural Properties of Disordered Protein States. *J. Phys. Chem. B* **2015**, *119*, 5113−5123.
(46) Jin, Z.; Du, X.; Xu, Y.; Deng, Y.; Liu, M.; Zhao, Y.; Zhang, B.; Li, X.; Zhang, L.; Peng, C. Structure of M pro from SARS-CoV-2 and discovery of its inhibitors. *Nature* **2020**, *582*, 289.
(47) Rackers, J. A.; Wang, Z.; Lu, C.; Laury, M. L.; Lagardère, L.; Schnieders, M. J.; Piquemal, J.-P.; Ren, P.; Ponder, J. W. Tinker 8: Software Tools for Molecular Design. *J. Chem. Theory Comput.* **2018**, *14*, 5273−5289.
(48) Lagardère, L.; Aviat, F.; Piquemal, J.-P. Pushing the Limits of Multiple-Time-Step Strategies for Polarizable Point Dipole Molecular Dynamics. *J. Phys. Chem. Lett.* **2019**, *10*, 2593−2599.
(49) McGibbon, R. T.; Beauchamp, K. A.; Harrigan, M. P.; Klein, C.; Swails, J. M.; Hernández, C. X.; Schwantes, C. R.; Wang, L.-P.; Lane, T. J.; Pande, V. S. MDTraj: A Modern Open Library for the Analysis of Molecular Dynamics Trajectories. *Biophys. J.* **2015**, *109*, 1528−1532.
(50) Pedregosa, F.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825−2830.
(51) Virtanen, P.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261−272.
(52) Brown, D. K.; Penkler, D. L.; Sheik Amamuddy, O.; Ross, C.; Atilgan, A. R.; Atilgan, C.; Tastan Bishop, O. MD−TASK: a software suite for analyzing molecular dynamics trajectories. *Bioinformatics* **2017**, *33*, 2768−2771.

https://doi.org/10.1021/acs.jpclett.1c01460
J. Phys. Chem. Lett. 2021, 12, 6218−6226

Conclusion

This study revealed that the M^{pro} undergoes long-range cooperative conformational changes between the dimerization interface and other cavities, resulting in what is known as allosteric interactions. PFFs were shown to be crucial for capturing these phenomena, which are purely long-range in nature. Furthermore, water molecules were found to play a critical role in the overall structural dynamics of the dimerization interface by participating in many-body polarization interactions with interface residues, contributing to the global allosteric effects between sites. As previously mentioned, accurate simulations of M^{pro} and other proteins require the simultaneous ability to describe all types of non-covalent interactions within the protein and solvent, as well as extensive sampling that goes beyond the microsecond time scale. Indeed, analyses of nPFF simulations conducted by the RIKEN center and DE Shaw research revealed a clear lack of sampling. These findings, combined with the previous study presented in the previous section, had significant implications in the design and synthesis of M^{pro} inhibitors.


THE JOURNAL OF
PHYSICAL CHEMISTRY
LETTERS
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY
July 8, 2021
Volume 12
Number 26
pubs.acs.org/JPCL
ACS Publications
Most Trusted. Most Cited. Most Read.
www.acs.org

3.3 A Novel Collective Variable-Free Multi-Level Enhanced Sampling Strategy for Accelerating Molecular Dynamics Simulations

Introduction

In recent years, CV-free methods have gained popularity for their ability to accelerate molecular dynamics simulations. Among these methods, Gaussian-accelerated Molecular Dynamics (GaMD) has shown particular promise due to its high sampling acceleration, user-friendly tunable parameters, and minimal additional computational cost. GaMD accelerates conformational sampling by adding a harmonic boost to the potential energy.

In this section, we present a novel multi-level enhanced sampling strategy designed for PFFs. To achieve this, a highly scalable GaMD implementation is combine with Tinker-HP GPU and additional enhanced sampling techniques. As a first speedup, we propose an extension of the GaMD formalism with a new mode that enables the use of flexible water models, such as AMOEBA, and multi time-step integrators. We then coupled this novel GaMD approach with Umbrella Sampling (US) and the unsupervised adaptive sampling method, as explained previously.

To demonstrate the applicability of these physics-based hybrid enhanced sampling strategies to PFFs, we performed Potential of Mean Force (PMF) calculations for a large biological complex, CD2-CD58, interacting via salt bridges with the AMOEBA force field.[173]

JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

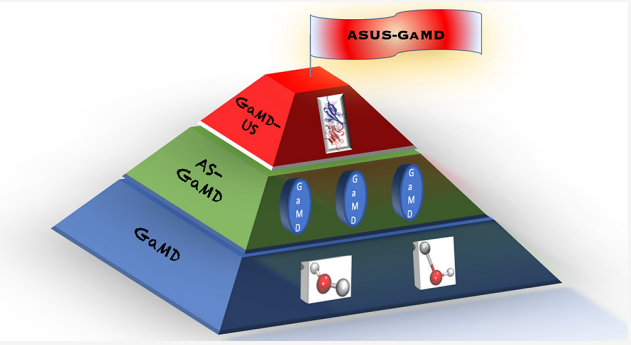
An Efficient Gaussian-Accelerated Molecular Dynamics (GaMD) Multilevel Enhanced Sampling Strategy: Application to Polarizable Force Fields Simulations of Large Biological Systems

Frédéric Célerse,[▽] Théo Jaffrelot Inizan,[▽] Louis Lagardère, Olivier Adjoua, Pierre Monmarché, Yinglong Miao, Etienne Derat, and Jean-Philip Piquemal*

Cite This: J. Chem. Theory Comput. 2022, 18, 968−977 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: We introduce a novel multilevel enhanced sampling strategy grounded on Gaussian-accelerated Molecular Dynamics (GaMD). First, we propose a GaMD multi-GPUs-accelerated implementation within the Tinker-HP molecular dynamics package. We introduce the new "dual-water" mode and its use with the flexible AMOEBA polarizable force field. By adding harmonic boosts to the water stretching and bonding terms, it accelerates the solvent−solute interactions while enabling speedups, thanks to the use of fast multiple−time step integrators. To further reduce the time-to-solution, we couple GaMD to Umbrella Sampling (US). The GaMD—US/dual-water approach is tested on the 1D Potential of Mean Force (PMF) of the solvated CD2−CD58 system (168 000 atoms), allowing the AMOEBA PMF to converge within 1 kcal/mol of the experimental value. Finally, Adaptive Sampling (AS) is added, enabling AS−GaMD capabilities but also the introduction of the new Adaptive Sampling−US−GaMD (ASUS−GaMD) scheme. The highly parallel ASUS−GaMD setup decreases time to convergence by, respectively, 10 and 20 times, compared to GaMD−US and US. Overall, beside the acceleration of PMF computations, Tinker-HP now allows for the simultaneous use of Adaptive Sampling and GaMD-"dual water" enhanced sampling approaches increasing the applicability of polarizable force fields to large-scale simulations of biological systems.



INTRODUCTION

Understanding interactions within biomolecules is crucial for many topics such as drug discovery. Some structural modifications, sometimes undetected by experiment, can drastically change the nature of the physics ruling interacting complex systems. For this reason, predicting the long timsecale conformational dynamics of proteins is a long-standing challenge within the conventional molecular dynamics (cMD) community.[1−6] It requires accurate models able to capture the true potential energy hyper-surface and long simulations to both access the large biological processes timescale and satisfy the ergodicity principle.[7] Therefore, accelerating MD has been a central field of research in the last decades.[8−11] Beside these developments, several additional strategies have been pursued over the years to further accelerate the simulations. They include the extensive use of high-performance computing (HPC) resources[4,12] and the optimization of GPU-accelerated modeling platforms.[13−15] Alternatively, an intensive algorithmic work has been undertaken, introducing techniques such as multiple-time-step integrator schemes[16,17] or collective variables-driven MD methods.[18,19] The latter have been found useful in enhanced sampling and free-energy calculation.[20−25] Although such methods are powerful, since they can estimate free energies of binding or the stability of secondary and quaternary structures of proteins,[26,27] the free-energy estimations can suffer from biases either generated by the initial choice of the collective variable (CV) or by the existence of multiple CV within the mechanism process (e.g., dual mechanisms).[28] For these reasons, collective variable−free methods have become increasingly popular.[29] Among them, the recent Gaussian-accelerated molecular dynamics (GaMD) has shown great promise, because of its high sampling acceleration, its user-friendly tunable parameters, and its minor additional computational cost.[30] GaMD accelerates conformational sampling by adding a harmonic boost to the potential energy. Coupled with the second-order cumulant expansion, GaMD allows us to

Received: October 12, 2021
Published: January 26, 2022

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

compute unbiased properties by using an accurate reweighting procedure through cumulant expansion to the second order.

Although new-generation many-body polarizable force field (PFFs) are more accurate in describing biomolecular interactions,[31−34] they are computationally more challenging than traditional approaches. Therefore, to overcome these limitations, here, we provide a novel general multilevel enhanced sampling strategy, which we apply to the PFF AMOEBA. To do so, we combine the Tinker−HP massively parallel multi-GPUs platform[15] together with a highly scalable GaMD implementation (*level 0*) and then additional enhanced sampling techniques based on recent developments of the field. As a first speedup, we propose an extension of the GaMD formalism with a new GaMD mode, enabling the use of flexible water models such as AMOEBA[35,36] and fast multiple-time-step integrators[17] (*level 1*). We then discuss the explicit coupling of such GaMD approach to Umbrella Sampling (US)[37] and Adaptive Sampling (AS)[6] techniques (*level 2*). To demonstrate their applicability to PFF, these physics-based hybrid enhanced sampling strategies are then applied to the Potential of Mean Force (PMF) study of a large biological complex CD2−CD58 interacting via salt bridges with the AMOEBA force field. Finally, we combine all of them together within the Adaptive Sampling−US−GaMD method (ASUS−GaMD) scheme (*level 3*).

METHOD: INTRODUCING THE GaMD "DUAL WATER" MODE

GaMD is a potential-biasing method for unconstrained enhanced sampling without the need to set a predefined CV. It smooths the potential energy surface by adding a harmonic boost potential, as described in the seminal paper.[11] Its general framework makes it suitable for the development of hybrid schemes and variants, such as replica-exchange umbrella sampling GaMD (GaREUS),[38] ligand GaMD (LiGaMD),[39] and peptide GaMD (Pep-GaMD).[40]

If the system potential energy is lower than a threshold energy E, a harmonic potential energy boost is applied to smooth the potential energy surface. By denoting $q \in \mathbf{R}^{3N}$ as the configurations when the system potential energy $U(q)$ is lower than a threshold energy E, a boost, which is dependent on $U(q)$ is added:

$$U'(q) = U(q) + \Delta U^{\mathrm{GaMD}}(U(q)) \tag{1}$$

with $\Delta U^{\mathrm{GaMD}}(U(q))$ being the external harmonic potential boost:

$$\Delta U^{\mathrm{GaMD}}(U(q)) = \begin{cases} 0 & U(q) \le E \\ \frac{1}{2}k(E - U(q))^2 & U(q) < E \end{cases} \tag{2}$$

and k the harmonic force constant. The two adjustable GaMD parameters k and E are automatically determined following the original procedure described in ref30. The boost intensity can be managed through a user-specified upper limit labeled as σ_0 (e.g., $10k_BT$) predefined before the simulation. To ensure accurate reweighting with the cumulant expansion the ΔU^{GaMD} standard deviation, $\sigma_{\Delta V}$, should satisfy $\sigma_{\Delta V} < \sigma_0$.[30,41,42] GaMD provides different modes: the boost is either applied on the total potential (GaMD−pot), on the dihedral potential (GaMD−dih), or on both at the same time (GaMD−dual).[43,44] Recently, another mode was introduced:LiGaMD, which adds the boost to a ligand nonbonded interactions,[39] accelerating the sampling of ligand−protein interactions. It is known that interactions involving water are essential for such systems and that protein stability processes are controlled by water−protein interactions.[6,45,46] To accelerate these interactions, one would like to use the GaMD−dual mode on the nonbonded interactions of water molecules. However, such a boost requires the evaluation of the complete nonbonded energies and, in the context of multitimestep integrators such as BAOAB−RESPA1,[17] where they are split between short range and long range, these are only available at the outer (large) time step. This type of integrators enables the use of larger time steps and thus a direct acceleration of MD. For example, the BAOAB−RESPA1 is based on a RESPA (Reference System Propagator Algorithm[16]) three-level splitting of forces (bonded, short-range nonbonded, and long-range nonbonded) within the Leimkuhler's BAOAB discretization of Langevin dynamics.[47] It allows up to a 7-fold acceleration for polarizable point dipole molecular dynamics.[17] But the fluctuations of the associated bias are such that it must be evaluated at shorter timesteps, so that the entire procedure is not compatible with multitimestep integrators such as BAOAB−RESPA1. For similar reasons, the GaMD−dual mode with a bias applied to the complete potential energy is not compatible with even simple RESPA integrators in which the potential energy is split between bonded and nonbonded terms. Therefore, GaMD−dual mode becomes rapidly limited by the simulation time. To overcome this issue, we developed a new mode, GaMD−dual-water (denoted as "GaMD−dualwat"), which adds a boost to the protein dihedral potential energy term and the water stretching and bending terms, this time fully compatible with RESPA and RESPA1 like integrators, allowing the water molecule to be more flexible, thus favoring their conformational changes.

$$\Delta U^{\text{GaMD-dw}}(U(q)) = \Delta U^{\mathrm{dihedral}}_{\mathrm{protein}}(U(q)) + \Delta U^{\mathrm{stretch}}_{\mathrm{water}}(U(q)) + \Delta U^{\mathrm{bend}}_{\mathrm{water}}(U(q)) \tag{3}$$

This mode is enabled by the flexibility of the AMOEBA 03 water model[35] but is not compatible with rigid water models, such as TIP3P,[48] commonly used with the CHARMM and AMBER force field.[49] This framework allows one to further reduce the computational cost gap between PFFs and nPFFs. This new mode, in addition to the other GaMD−dih and GaMD−dual modes, is now available within the Tinker−HP software.[12,15] In the following, we first tested its GPU scalability and performance on the STMV system (∼1 066 624 atoms), and its sampling efficiency is demonstrated on simulations of the alanine dipeptide and the CD2-CD58 complex. A technical appendix is present at the end of the manuscript and provides the formalism of the method and the associated debiasing equations.

RESULTS AND DISCUSSION

Level 0: Efficiency and GPU Scalability. The GaMD implementation is such that only a small computational and communication (in parallel) overhead is added, compared to cMD. The GaMD−dih and GaMD−dualwat have been considered on the STMV system (1 066 624 atoms) with the AMOEBA PFF and the 10 fs outer time-step HMR BAOAB−RESPA1 multiple-time-step integrator.[17] V100 GPUs from the national Jean Zay supercalculator have been used for all the

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

benchmark computations. Similar scalability studies have been performed on the Jean Zay multi-CPUs (Figure S1 in the Supporting Information). The AMOEBA GPU simulations were performed on a single node, since the multinode extension of the AMOEBA PFF within the Tinker−HP package is still under development. On 1 and 2 GPUs (Figure 1), the GaMD data communications are negligible (1%). On 4

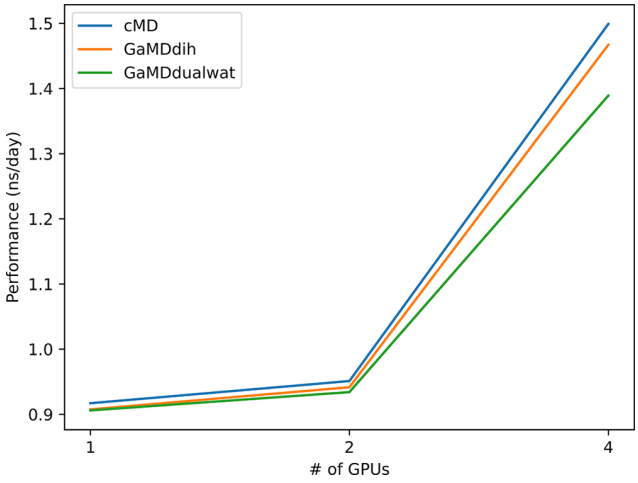


Figure 1. GaMD−dih and GaMD−dualwat scaling performance on 1/2/4 V100 GPUs (i.e., corresponding to a full node of the Jean Zay machine) on STMV (1 066 624 atoms) with the AMOEBA force field and the BAOAB−RESPA1 10 fs multiple-time-step integrator. The cMD reference, in blue, allows one to evaluate the GaMD impacts on the code communications.

GPUs, the communications are increasing and the performance decreases by 7%. Overall, the use of GaMD only slightly alters the performance. This high scalability opens the door to simulate at a high accuracy, large complex biomolecular systems with PFFs.

Level 1: GaMD−dualwat with PFFs. We compared GaMD−dih, GaMD−dual, and GaMD−dualwat sampling acceleration on the exploration of the relevant basins of the alanine dipeptide (e.g., α_r, α_L, and P_{II}). The alanine dipeptide is solvated in a cubic 20 Å water box. We used the many-body AMOEBABIO18 PFF.[50,51] The system was minimized with a RMS of 1 kcal/mol and sampled within the NPT thermodynamic ensemble with the Bussi thermostat[52] and a MonteCarlo barostat[53] at 300 K and 1 atm. We used the Velocity Verlet integrator and a 1 fs time step.[54] The Smooth Particle Mesh Ewald (SPME) algorithm was employed to compute noncovalent interactions[55] with a real space cutoff equal to 7 Å and a van der Waals cutoff set to 9 Å. For AMOEBA, the convergence criteria for multipoles was set to 10^{-5}. After short testing simulations, we found an optimal value of 3 kcal/mol for GaMD−dih and GaMD−dual σ_0, in accordance with ref 11, and 4 kcal/mol for GaMD−dualwat (see Figure S2 and Tables S1 and S2 in the Supporting Information). We ran three independent simulations of 60 ns for each mode. The different sampled basins are also compared to a 1 μs cMD AMOEBA reference.

Reweighted, see the Technical Appendix, free-energy surfaces obtained from these simulations are depicted in Figure 2 and show that GaMD−dual captures the α_r (50°,25°), α_L (−75°,−25°), and P_{II} (−75°,150°) basins well. These results are consistent with the 1 μs cMD trajectory (see Figure S3 in the Supporting Information) depicting these three basins. While the GaMD−dih mode captures the α_r basin after 150 ns, the GaMD−dualwat captures it in 100 ns (see Figure S4 in the Supporting Information). We also observe a sampling acceleration between GaMD−dual and GaMD−dualwat compared to the ref 1 microsecond cMD. To characterize the GaMD boost harmonicity, its distribution anharmonicity (γ) is calculated as in ref 30. γ serves as an indicator of the sampling convergence and reweighting procedure accuracy. Depicted in Figure S5 in the Supporting Information, GaMD−dih as well as GaMD−dual depicts high anharmonicity with, respectively, 0.252 and 0.016, compared to GaMD−dualwat with 0.0005. In addition, we see a steep anharmonicity convergence to less than 10^{-3} for GaMD−dualwat while being relatively stable at 2×10^{-1} for GaMD−dih (see Figure S4). In comparison the anharmonicity is ∼0.001 with GaMD−dih and AMBER99SB. Therefore, PFFs increase the statistical noise and stress the importance of using low-anharmonicity GaMD modes. In that sense, GaMD−dualwat appears more suitable than GaMD−dual for PFFs simulations with an anharmonicity equal to 0.0005. As stated previously, another advantage of GaMD−dualwat is that it can be coupled to multiple-time-step procedures, such as BAOAB−RESPA1,[17] in contrast to the GaMD−dual mode, which remains limited to single-time-step integrators. Comparative results of GaMD−dualwat with both integrators can be found in Figure S6 in the Supporting Information. Its coupling with multiple-time-step procedures clearly compensates for the slightly lower sampling performance, compared with GaMD−dual. The sampling enhancement brought by the GaMD−dualwat can be partly related to how it affects the diffusion of water: in Table S3 of the Supporting Information, we report the self-diffusion coefficients of bulk water computed within a same setup (same size of box and same integrator) and observe that it is increased with the GaMD−dualwat mode, compared to the simple GaMD−dih one, favoring global conformational changes due to water reorganization. While the added sampling efficiency is already significant for the alanine dipeptide, we expect it to be larger on more complex and larger biological systems such as CD2CD58, where water reorganization plays a bigger role.

Combined with a highly parallel GPUs infrastructures and multiple-time-step integrators, the GaMD−dualwat should allow one to help reach very high-resolution conformational space of large molecular systems. In addition to the sampling acceleration it provides, the low associated anharmonicity drastically reduces the statistical noise associated with reweighting.

Level 2: Accelerating Simulations with the Parallel AS−GaMD Scheme. We further coupled our newly introduced GaMD mode to additional enhanced sampling strategies. Recently, we developed a new adaptive sampling (AS) technique, which was shown to allow massive sampling of the SARS−CoV−2 Main Protease conformational space.[6] We coupled these two methodologies together, yielding the AS−GaMD method. The principle is similar to the AS, the only modification being that each cMD at each iteration is now a GaMD simulation. The double bias coming from both AS and GaMD implies that a suitable and careful reweighting scheme must be introduced to reconstruct an unbiased free-energy surface. All mathematical tools for the reweighting scheme are provided in the Technical Appendix. We applied this methodology to the same system, the alanine dipeptide, using the same GaMD simulation protocol. At each iteration,

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

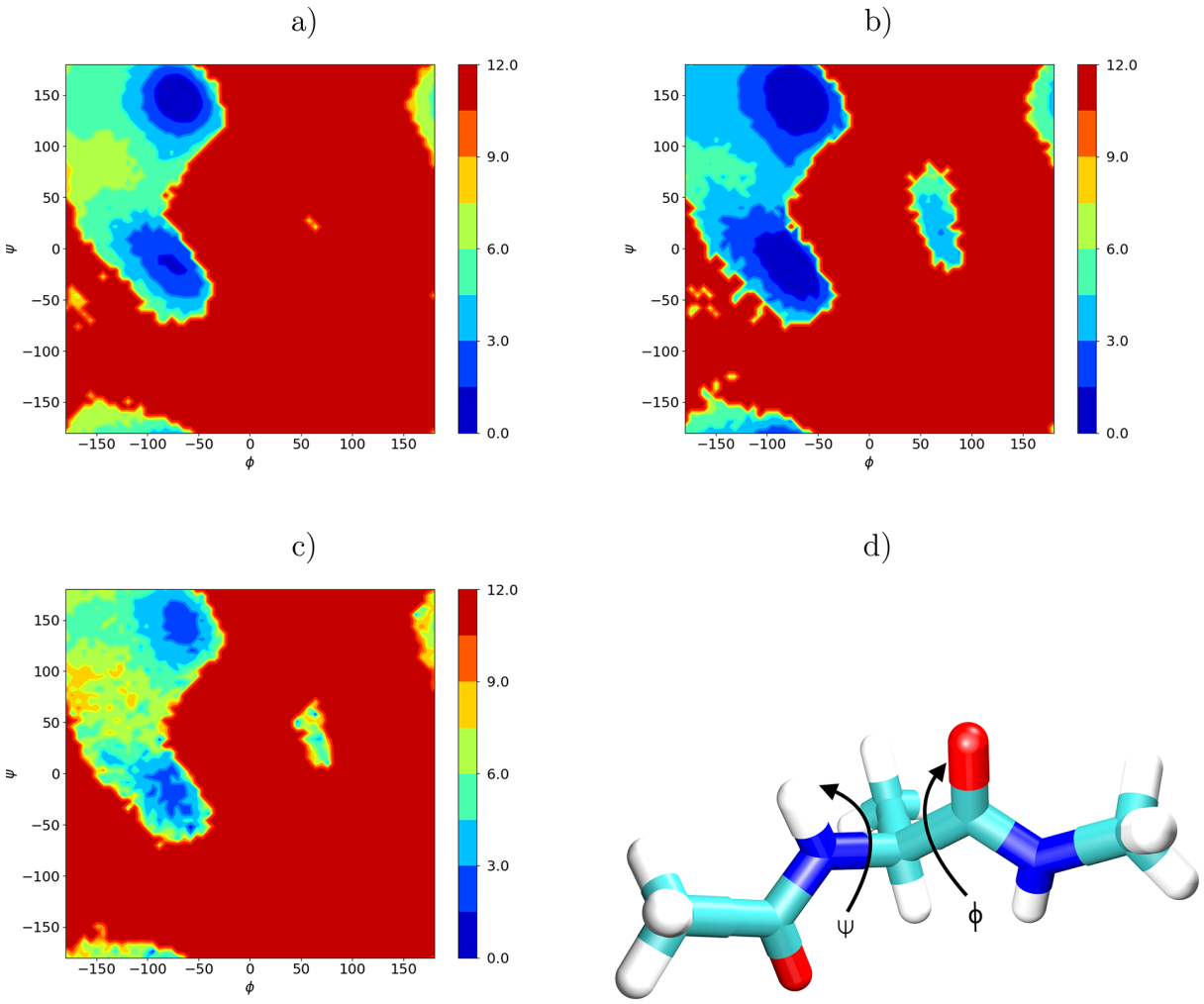


Figure 2. 2D PMF (in kcal/mol) of the alanine dipeptide obtained in AMOEBA for (a) GaMD−dih mode (3 × 60 ns), (b) GaMD−dual mode (3 × 60 ns), and (c) GaMD−dualwat mode (3 × 60 ns). (d) Alanine dipeptide representation with the corresponding Φ and Ψ angles.

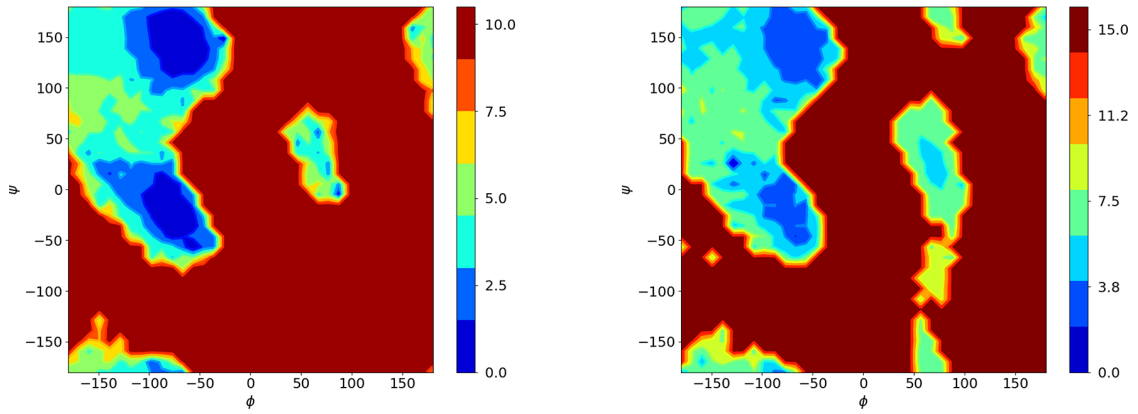


Figure 3. PMFs of (a) GaMD dualwat (3 × 60 = 180 ns) and (b) AS−GaMD dualwat (5 × 25 = 125 ns) simulations for the alanine dipeptide. For the GaMD dualwat simulations, we performed three independent simulations of 60 ns using 1 fs time step with the verlet integrator. The AS−GaMD dualwat simulations were performed using the BAOAB−RESPA1 10 fs multi-time-step integrator and 5 AS iterations of 5 × 5 ns with a square term.

we projected the structures on the two main dihedral angle spaces. To push the limit of the AS−GaMD sampling capability, we combined a modified version of the AS selection scheme with the BAOAB−RESPA1 multi-time-step integrator. The probability law for selection of new structures was taken as the inverse of the square of the probability density on the reduced space, which further amplifies the exploration of undiscovered region.

In Figure 3, we represented the 2D PMF obtained with both AS−GaMD/BAOAB−RESPA1 and GaMD/VERLET simula-

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

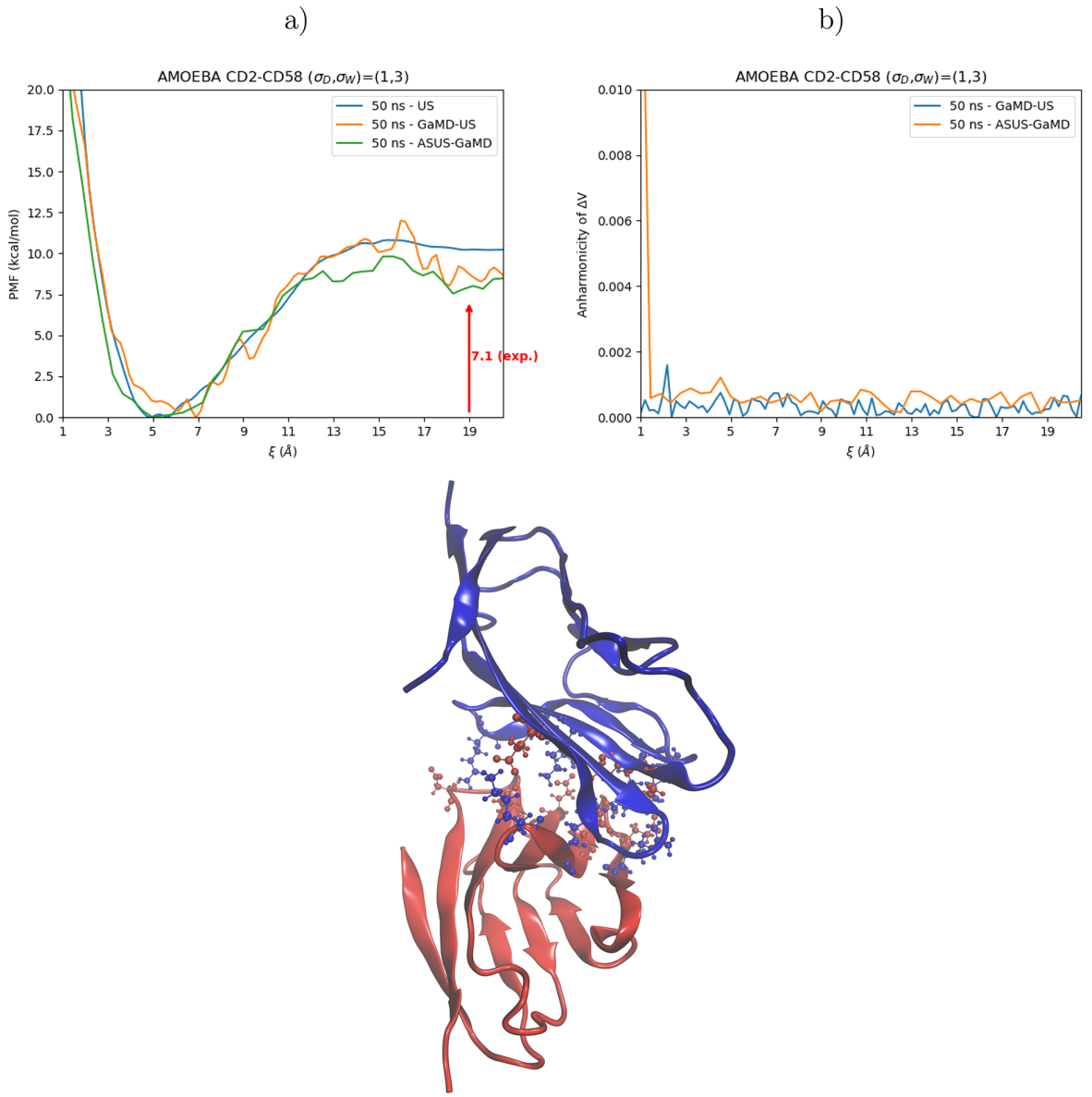


Figure 4. (c) PDB 1QA9 CD2CD58 representation with CD2 and CD58 subcomplexes represented respectively in blue and red, using the new ribbons representation. Residues at the interface considered in the COM distance between the two subcomplexes are represented in blue and red for respective basic and acid residues using the CPK representation. VMD software was employed to generate the structure. PMFs obtained with US, GaMD−US, and ASUS−GaMD are depicted in panel (a) and their respective anharmonicity in panel (b).

tions. For AS−GaMD/BAOAB−RESPA1, we performed 5 iterations of 5 × 5 ns GaMD−dualwat simulations for a total simulation time of 125 ns. As in the previous section, the GaMD/VERLET is composed of three independent simulations of 60 ns (180 ns total). In 30% less simulation time and 5 times less computational time, thanks to the natural AS parallelism, the coupled AS−GaMD/multi-time-step integrator scheme greatly enhances the exploration of the free-energy surface. We observed that the α_L region is already captured at the first iteration, i.e., with only 25 ns (see Figure S7 in the Supporting Information). In addition, other states, next to the α_L region, are captured within tens of nanoseconds and are still not seen after the entire GaMD simulation. Therefore, this AS−GaMD/multi-time-step coupling can represent an important gain for the sampling of biomolecular systems.

Level 3: Pushing the Limit of PMF Convergence with GaMD−US and ASUS−GaMD. US has been widely used and is mathematically robust but it is still suffers from several issues.[56−58] In addition to the choice of the CVs, it is also difficult to estimate the PMF convergence, since it is system-dependent. Good indicators to check if convergence is reached are the overlap between neighboring windows and the evolution of the PMF curve, as a function of the simulation time per window. To accelerate the sampling within each window, Oshima et al. recently combined GaMD with replica-exchange and US.[38] Here, we first only applied a GaMD boost in each US window in order to enhance the sampling in the orthogonal space.

To demonstrate the PMF convergence acceleration, we studied the dissociation of the salt bridges interface within the CD2CD58 complex. This system, made of several salt bridges

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

and hydrogen bondings interactions, was already studied by some of us.[28] Although it has been shown that PFFs allow a better description of the salt bridges interactions, their computational cost has long hindered the study of such a large system. Since the portability of Tinker-HP on multi-GPU and the global acceleration of the PFFs, reaching such system is now easily achievable. To start this study, we took the same CD2CD58 complex as in our previous work[28] but we solvated it in a waterbox of 100 Å × 100 Å × 100 Å. Counterion were added to neutralize the system. We used the AMOEBABIO18 PFF.[50,51] The system was minimized with a RMS of 1 kcal/mol in the NVT thermodynamic ensemble with the Bussi thermostat.[52] Temperature was set to 300 K while pressure was set to 1 atm. We used the multi-time-step BAOAB−RESPA1 with a 10 fs time step with the Hydrogen Mass Repartitioning scheme (HMR)[17] and Smooth Particle Mesh Ewald (SPME) algorithm to compute electrostatic and polarization interactions[55] with a real-space cutoff of 7 Å and a van der Waals cutoff of 9 Å. The convergence criteria for polarization was set to 10^{-5}. Thirty nine (39) US windows were generated, ranging from 1 to 20 Å with a width of 0.5 Å between them. CV was chosen as the distance between the center of mass formed by the interfacial residues isolated by Bayas et al. on CD2 and CD58 (see Table 1 in ref 59). A spring constant of 10 kcal/(mol Å^2) was employed to restrain the system along the chosen CV. Each window was run for 5 ns for equilibration and then for 50 ns. Histogram overlap as well as the PMF curve, as a function of the simulation time allocated per window, were employed to check the convergence of the simulations (see Figure S9 in the Supporting Information). The final US PMF shows a slow decrease of the free-energy barrier with the simulation time, suggesting a slow convergence to ∼12.5 kcal/mol. Binding affinity was found to be, experimentally, ∼7.1 ± 0.03 kcal/mol, suggesting that our simulations are not converged.[59] In order to improve sampling within each window, a new US procedure was performed, similar to the previous US protocol, but now with an additional GaMD−dualwat potential applied in each window. The GaMD parametrization protocol and reweighting procedure are described in the Technical Appendix and in Figure S8 and Table S4 in the Supporting Information). The optimized GaMD−dualwat parameter (σ_0) values are equal to 1 and 3 kcal/mol for the dihedral and dual water modes, respectively. Figure 4 shows the difference between standard US and GaMD−US. The GaMD−US PMF and boost harmonicity converge at 40 ns per window (see Figures S10A−S10C in the Supporting Information). The predicted free-energy barrier is now within the 1 kcal/mol of the experiment. It shows that GaMD−dualwat, even without the presence of replica exchange, could considerably improve the PMF convergence of large systems. It also demonstrates that salt bridges and, more generally, protein−protein interactions are well-described with PFFs. Furthermore, as demonstrated in the work of Debiec et al.,[60] the improved accuracy of non-PFFs in describing these interactions requires the implicit incorporation of solvent polarization, underscoring the importance of polarization effects in these contexts.

To further push the sampling, we coupled together GaMD, AS, and US (ASUS−GaMD). We provide two reweighting schemes that either use modified Multistate Bennett Acceptance Ratio (MBAR) equations or the Rao−Blackwell estimator. The mathematical expressions are general and can be used with any weighted dynamics. Starting from an initial US simulation (approximately equal to a few nanoseconds), each window is decomposed in several AS independent trajectories with an additional GaMD−dualwat potential boost (GaMD). Here, we ran 2 iterations of 5 × 5 ns GaMD−US per window. The PMF evolution can be found in Figure S11 in the Supporting Information, while the resulting PMF is depicted in Figure 4. We observe that ASUS−GaMD reaches GaMD−US in one iteration, showing the sampling acceleration impact provided by the AS part within ASUS−GaMD. Note that the rough aspect of the PMF obtained with the methods involving a GaMD bias comes from the debiasing of the boost potential, as can be seen in previous work involving US and GaMD.[38] Although a careful reweighting is needed for the different AS, GaMD and US layers, the overall ASUS−GaMD approach inherits the strong adaptive sampling advantages of being pleasantly parallelizable and considerably accelerates the PMF convergence.

CONCLUSIONS

Combined with the use of modern GPUs, these sampling techniques allow one to drastically reduce time to solution in PFFs evalution of PMFs. Although it is difficult to truly quantify the final acceleration (i.e., a PMF convergence remains partially system-dependent), one can see in Figure S12 in the Supporting Information that if we extrapolate the US convergence, ASUS−GaMD converges 1.4 times faster. Thanks to the native parallelism inherited from AS, the PMF evaluation can be done in one-fifth of the simulation time, yielding an acceleration of 7. If we consider that convergence was already reached with a 25 ns per window setup, this factor grows to 14. Thus, ASUS−GaMD that would have taken months can be reduced to days of computation. This work also allows one to invoke any variant of the combined approaches, thereby offering access to GPU-accelerated GaMD-adaptive sampling (AS−GaMD) simulations that will be helpful to further extend conformational space studies of proteins[6] To conclude, these methodologies will contribute further to allow high-resolution sampling of large biological systems up to millions of atoms, using a polarizable force field.

TECHNICAL APPENDIX

We use $\xi(q)$ to denote the reaction coordinate along which we performed the US simulation and q is the configuration. Here, a configuration means the positions $q \in \mathbf{R}^{3N}$ of all the atoms of the system. The imposed US bias potential is

$$U_j^{\mathrm{US}}(q) = \mathcal{K}(\xi(q) - \xi_j)^2 \tag{A1}$$

with $\mathcal{K}$ being the force constant.

We combined the AS, US, and GaMD such that each US window $j \in [[1, ..., M]]$, $\xi_1, ..., \xi_M$, is parallelized and accelerated by adaptive sampling replicas and GaMD boost potential:

$$U_j''(q) = U(q) + U^{\mathrm{GaMD}}(q) + U_j^{\mathrm{US}}(q) \tag{A2}$$

We use $(q_{j,n})_{n\in1,N}$ to denote the N configurations generated by the AS replicas of US window j and $(\omega_{j,n})_{n\in1,N}$ to represent their respective AS weights. These normalized weights are defined as $\omega_{j,n} = \frac{N v_{j,n}}{\sum_{m=1}^{N} v_{j,m}}$ so that $\sum_{n=1}^{N} \omega_{j,n} = N$ with $v_{j,n}$ being the unnormalized AS weights. The canonical average of an observable φ is estimated by

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

$$\langle\varphi\rangle_j'' = \frac{\int \varphi(q) e^{-\beta U_j''(q)}\,\mathrm{d}q}{\int e^{-\beta U_j''(q)}\,\mathrm{d}q} \simeq \frac{\sum_{n=1}^{N}\varphi(q_{j,n})\omega_{j,n}}{\sum_{n=1}^{N}\omega_{j,n}} = \frac{1}{N}\sum_{n=1}^{N}\varphi(q_{j,n})\omega_{j,n} \tag{A3}$$

In practice, to get a smooth reweighted PMF, the reaction coordinate ξ is discretized in K bins around values $x_1, ..., x_K$. We want to estimate for each $k \in [\![1, ..., K]\!]$ its free energy, up to an additive constant,

$$F(x_k) = -\frac{1}{\beta}\ln \mathbf{P}(\xi(q) \in \mathrm{Bin}(x_k)) \tag{A4}$$

where q is distributed according to the density probability law, $\frac{e^{-\beta U}}{\int e^{-\beta U}}$, i.e.,

$$F(x_k) = -\frac{1}{\beta}\ln \frac{\int 1_{\xi(q)\in \mathrm{Bin}(x_k)} e^{-\beta U(q)}\,\mathrm{d}q}{\int e^{-\beta U(q)}\,\mathrm{d}q} = -\frac{1}{\beta}\ln\langle\varphi_k\rangle \tag{A5}$$

with $\varphi_k = 1_{\xi(q)\in \mathrm{Bin}(xk)}$.

First Step: GaMD with Cumulant Expansion. We, first, remove the GaMD bias. Here, we want to find a relationship between $\langle\varphi\rangle$ and $\langle\varphi\rangle'$, where the prime average represents the canonical average over the potential $U' = U + U^{\mathrm{GaMD}}$. Starting from the canonical average, we notice

$$\begin{aligned}\langle\varphi\rangle &= \frac{\int \varphi(q) e^{-\beta U(q)}\,\mathrm{d}q}{\int e^{-\beta U(q)}\,\mathrm{d}q} \\ &= \frac{\int \varphi(q) e^{\beta U^{\mathrm{GaMD}}(q)} e^{-\beta U'(q)}\,\mathrm{d}q}{\int e^{\beta U^{\mathrm{GaMD}}(q)} e^{-\beta U'(q)}\,\mathrm{d}q} \\ &= \frac{\langle\varphi e^{\beta U^{\mathrm{GaMD}}}\rangle'}{\langle e^{\beta U^{\mathrm{GaMD}}}\rangle'}\end{aligned} \tag{A6}$$

By applying this with $\varphi = \varphi_k$,

$$\begin{aligned}F(x_k) &= -\frac{1}{\beta}\ln\frac{\langle\varphi_k e^{\beta U^{\mathrm{GaMD}}}\rangle'}{\langle e^{\beta U^{\mathrm{GaMD}}}\rangle'} = -\frac{1}{\beta}\ln\langle\varphi_k e^{\beta U^{\mathrm{GaMD}}}\rangle' + C \\ &= F'(x_k) - \frac{1}{\beta}\ln\frac{\langle\varphi_k e^{\beta U^{\mathrm{GaMD}}}\rangle'}{\langle\varphi_k\rangle'} + C\end{aligned} \tag{A7}$$

where C is a constant and $F'(x_k)$ is the free energy, $F'(x_k) = -\frac{1}{\beta}\ln\langle\varphi_k\rangle'$. To reduce the estimator variance, we used the cumulant expansion to the second order,

$$\begin{aligned}\ln\frac{\langle\varphi_k e^{\beta U^{\mathrm{GaMD}}}\rangle'}{\langle\varphi_k\rangle'} \simeq{}& \beta\frac{\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{\langle\varphi_k\rangle'} \\ &+ \frac{\beta^2}{2}\left(\frac{\langle\varphi_k (U^{\mathrm{GaMD}})^2\rangle'}{\langle\varphi_k\rangle'} - \left(\frac{\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{\langle\varphi_k\rangle'}\right)^2\right)\end{aligned} \tag{A8}$$

By combining with eq A7, the free energy is rewritten as

$$\begin{aligned}F(x_k) \simeq{}& -\frac{1}{\beta}\ln\langle\varphi_k\rangle' - \beta\frac{\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{\langle\varphi_k\rangle'} \\ &- \frac{\beta^2}{2}\left(\frac{\langle\varphi_k (U^{\mathrm{GaMD}})^2\rangle'}{\langle\varphi_k\rangle'} - \left(\frac{\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{\langle\varphi_k\rangle'}\right)^2\right) + C\end{aligned} \tag{A9}$$

Second Step: AS Modified MBAR. Finally, we want to express $\langle\varphi\rangle'$, with respect to the AS weights in each US window $j \in [\![1, ..., M]\!]$. This can be done in two ways, using either the MBAR or the Rao−Blackwell estimator.

Modified MBAR. Let us define c_j' and F_j' as

$$c_j' = \int e^{-\beta U_j''(q)}\,\mathrm{d}q \qquad F_j' = -\frac{1}{\beta}\ln c_j' \tag{A10}$$

The prime comes from the use of the MBAR on the reference energy U' of the previous section. The starting point is to use the MBAR identity (eq 5 in ref 61) and notice

$$c_i'\langle e^{\beta U_j''}\alpha_{i,j}\rangle_i'' = \int e^{-\beta U_j''(q)} e^{-\beta U_i''(q)}\alpha_{i,j}(q)\,\mathrm{d}q = c_j'\langle e^{\beta U_i''}\alpha_{i,j}\rangle_j'' \tag{A11}$$

which holds for arbitrary functions $q \to \alpha_{ij}(q)$ with $i, j \in [\![1, ..., M]\!]$. Notice that each window generated the same number of configurations N. The MBAR estimator has been proven to be optimal by using

$$\alpha_{i,j}(q) = \frac{1/c_j'}{\sum_{k=1}^{M} e^{-\beta U_k''(q)}/c_k'} \tag{A12}$$

and by summing over j:

$$c_i'\sum_{j=1}^{M}\left\langle\frac{e^{-\beta U_j''}/c_j'}{\sum_{k=1}^{M} e^{-\beta U_k''}/c_k'}\right\rangle_i'' = \sum_{j=1}^{M} c_j'\left\langle\frac{e^{-\beta U_i''}/c_j'}{\sum_{k=1}^{M} e^{-\beta U_k''}/c_k'}\right\rangle_j'' \tag{A13}$$

We obtain a set of M equations for all $i \in [\![1, ..., M]\!]$

$$c_i' = \sum_{j=1}^{M}\left\langle\frac{e^{-\beta U_i(q)}}{\sum_{k=1}^{M} e^{-\beta U_k(q)}/c_k'}\right\rangle_j'' \tag{A14}$$

Using eq A3 we obtain the estimators

$$\hat{c}_i' = \frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\frac{\omega_{j,n} e^{-\beta U_i(q_{j,n})}}{\sum_{k=1}^{M} e^{-\beta U_k(q_{j,n})}/\hat{c}_k'} \tag{A15}$$

and finally with eq A10

$$F_i' = -\frac{1}{\beta}\ln\left(\frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\frac{\omega_{j,n} e^{-\beta U_i(q_{j,n})}}{\sum_{k=1}^{M} e^{\beta(F_k' - U_k(q_{j,n}))}}\right) \tag{A16}$$

which must be solve self-consistently.

Modified Rao−Blackwell Estimator. Recently, Ding et al.[62,63] derived the MBAR equations using the Rao−Blackwell (RB) estimator. The RB theorem characterizes the transformation of a crude estimator into a better estimator that has smaller mean squared error, with respect to the dataset.

We wish to calculate the $i \in [\![1, ..., M]\!]$ relative free energies F_i^* of M thermodynamic states sampled independently, with potential U_i. To compute the relative free energies, the system should be sampled according to the Boltzmann distribution. We note $q_{i,n}$ with the $n \in [\![1, ..., N_i]\!]$ configurations sampled from state i. To compute the relative free energies of the M thermodynamic states, the configurations $q_{i,n}$ are combined and considered as samples from the generalized ensemble $p_i(q) \propto e^{-\beta(U_i(q)+b_i)}$, where b_i is an unknown biased energy. This biased energy was introduced[62] to adjust the relative weight of state i to be proportional to N_i, leading to

https://doi.org/10.1021/acs.jctc.1c01024
J. Chem. Theory Comput. 2022, 18, 968−977

$$F_i = F_i^* + b_i = -\frac{1}{\beta}\ln\frac{N_i}{N} \tag{20}$$

where $N = \sum_{i=1}^{M} N_i$. From this equation, we can then use the RB estimator

$$F_i = -\frac{1}{\beta}\ln p_i = -\frac{1}{\beta}\ln\frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N} p_i(q_{j,n})$$

$$= -\frac{1}{\beta}\ln\frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\frac{\omega_{j,n}e^{-\beta(U_i(q_{j,n})+b_i)}}{\sum_{k=1}^{M}e^{-\beta(U_k(q_{j,n})+b_i)}} \tag{A18}$$

Combining with eq 20:

$$1 = \frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\frac{\omega_{j,n}e^{-\beta(U_i(q_{j,n})+b_i)}}{\sum_{k=1}^{M}e^{-\beta(U_k(q_{j,n})+b_i)}} \tag{A19}$$

Thus, the unbiased free energy F_i^* can be calculated using eq 20 after solving eq A19 for b_i. Equation A19 has major interests: (1) it is more stable, (2) it reduces the number of floating point operations, and (3) the problem is reduced to minimizing a convex function. Indeed, if we define

$$g_i(b_1, ..., b_M) = \frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\frac{\omega_{j,n}e^{-\beta(U_i(q_{j,n})+b_i)}}{\sum_{k=1}^{M}e^{-\beta(U_k(q_{j,n})+b_i)}} - 1 \tag{A20}$$

then solving eq A19 is equivalent to finding the zeros of $(g_1, ..., g_M)$. Moreover, we can remark that the function $g_i = \nabla_{b_i} f$ where the function f is given by

$$f(b_1, ..., b_M) = -\frac{1}{N}\sum_{j=1}^{M}\sum_{n=1}^{N}\omega_{j,n}\ln\left(\sum_{k=1}^{M}e^{-\beta(U_k(q_{j,n})+b_i)}\right) - \sum_{j=1}^{M} b_j' \tag{A21}$$

which means solving eq A19 is equivalent to finding the critical points of f. Ding et al. have shown that f is convex, so the problem is reduced to minimizing this function which can be done with the L-BFGS method. The reweighting procedure, which uses part of the FastMBAR code, takes a few minutes on a single GPU. In this work we used the latter procedure, thanks to its GPU efficiency.

Third Step: ASUS-GaMD Reweighting. With either using the MBAR or the RB estimator procedure, we can extract the still-biased free energies. The final step is to derive an expression of $\langle\varphi\rangle'$, with respect to either $\hat{c}_k'$ or F_k'. By setting $c_0 = \int e^{-\beta U'(q)}\,dq$ and using (A3),

$$c_0\langle\varphi\rangle' = \int\varphi(q)e^{-\beta U'(q)}\,dq$$

$$= \sum_{i=1}^{M}\frac{\int\varphi(q)e^{-\beta U'(q)}e^{-\beta U_i''(q)}/c_i'\,dq}{\sum_{j=1}^{M}e^{-\beta U_j''(q)}/c_j'}$$

$$= \sum_{i=1}^{M}\left\langle\frac{\varphi}{\sum_{j=1}^{M}e^{-\beta U_j(q)}/c_j'}\right\rangle_i''$$

$$\simeq \frac{1}{N}\sum_{i=1}^{M}\sum_{n=1}^{N}\frac{\varphi(q_{i,n})\omega_{i,n}}{\sum_{k=1}^{M}e^{-\beta U_k(q_{j,n})}/\hat{c}_k'} \tag{A22}$$

in other words,

$$c_0\langle\varphi\rangle' \simeq \frac{1}{NM}\sum_{i=1}^{M}\sum_{n=1}^{N}\varphi(q_{i,n})r_{i,n} \tag{A23}$$

with $r_{i,n}$ the weight of configuration $q(i, n)$:

$$r_{i,n} = \frac{M\omega_{i,n}}{\sum_{k=1}^{M}e^{-\beta U_k(q_{i,n})}/\hat{c}_k'} \tag{A24}$$

c_0 is unknown but is not dependent on $k \in [[1, ..., K]]$ so eq A9 can be rewritten as

$$F(x_k) \simeq -\frac{1}{\beta}\ln\langle c_0\varphi_k\rangle' - \beta\frac{c_0\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{c_0\langle\varphi_k\rangle'}$$

$$-\frac{\beta^2}{2}\left(\frac{c_0\langle\varphi_k(U^{\mathrm{GaMD}})^2\rangle'}{c_0\langle\varphi_k\rangle'} - \left(\frac{c_0\langle\varphi_k U^{\mathrm{GaMD}}\rangle'}{c_0\langle\varphi_k\rangle'}\right)^2\right) + C' \tag{A25}$$

with

$$c_0\langle\varphi_k\rangle' \simeq \frac{1}{NM}\sum_{i=1}^{M}\sum_{n=1}^{N} r_{i,n}\mathbf{1}_{\xi(q_{i,n})\in\mathrm{Bin}(x_k)}$$

$$c_0\langle\varphi_k U^{\mathrm{GaMD}}\rangle' \simeq \frac{1}{NM}\sum_{i=1}^{M}\sum_{n=1}^{N} U^{\mathrm{GaMD}}(q_{i,n})r_{i,n}\mathbf{1}_{\xi(q_{i,n})\in\mathrm{Bin}(x_k)} \tag{A26}$$

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.1c01024.

CPUs scalability of GaMD (Figure S1); parametrization of GaMD parameters for the alanine dipeptide (Figure S2, Tables S1 and S2); free-energy surface of alanine dipeptide obtained from a long cMD simulation (Figure S3); the GaMD influence on the water diffusion property (Table S3); evolution of anharmonicity and basins's populations, as a function of the simulation time for the alanine dipeptide (Figure S4); boost distribution and anharmonicity from AMOEBA GaMD simulations on the alanine dipeptide (Figure S5); GaMD applied with Velocity Verlet and BAOAB−RESPA1 multi−time−step integrator (Figure S6); evolution of the AS−GaMD sampling, as a function of the simulation time (Figure S7); parametrization of GaMD parameters for the CD2−CD58 (Figure S8 and Table S4); U.S. convergence on CD2−CD58 (Figure S9); GaMD−US convergence on CD2−CD58 (Figure S10); ASUS−GaMD convergence on CD2−CD58 (Figure S11); US, GaMD−US, and ASUS−GaMD extrapolated convergence (Figure S12) (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Jean-Philip Piquemal − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France; The University of Texas at Austin, Department of Biomedical Engineering, Austin, Texas 78705, United States; Institut Universitaire de France, Paris 75005, France;* orcid.org/0000-0001-6615-9426; Email: jean-philip.piquemal@sorbonne-universite.fr

Authors

Frédéric Célerse − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France; IPCM, UMR 8232 CNRS, Sorbonne Université, Paris 75005, France;* Present Address: LCMD EPFL, CH-1015 Lausanne, Switzerland

Théo Jaffrelot Inizan − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France*

Louis Lagardère − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France; IP2CT, FR 2622 CNRS, Sorbonne Université, Paris 75005, France*

Olivier Adjoua − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France*

Pierre Monmarché − *LCT, UMR 7616 CNRS, Sorbonne Université, Paris 75005, France; LJLL, UMR 7598 CNRS, Sorbonne Université, Paris 75005, France*

Yinglong Miao − *Center for Computational Biology and Department of Molecular Biosciences, University of Kansas, Lawrence, Kansas 66045, United States;* orcid.org/0000-0003-3714-1395

Etienne Derat − *IPCM, UMR 8232 CNRS, Sorbonne Université, Paris 75005, France*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.1c01024

Author Contributions

[∇]These authors contributed equally to this work.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work has received funding from the European Research Council (ERC), under the European Union's Horizon 2020 Research and Innovation Program (Grant Agreement No. 810367), Project EMC2 (JPP). F.C. is thankful for funding from the French state funds managed by the CalSimLab LABEX and the ANR within the Investissements d'Avenir program (Reference No. ANR11-IDEX-0004-02) and support from the Direction Génerale de l'Armement (DGA) Maîtrise NRBC of the French Ministry of Defense. Computations have been performed at GENCI (IDRIS, Orsay, France and TGCC, Bruyères le Chatel) on Grant No. A0070707671.

■ REFERENCES

(1) Caves, L. S.; Evanseck, J. D.; Karplus, M. Locally accessible conformations of proteins: multiple molecular dynamics simulations of crambin. *Protein Sci.* **1998**, *7*, 649−666.

(2) Schlitter, J.; Engels, M.; Krüger, P. Targeted molecular dynamics: a new approach for searching pathways of conformational transitions. *J. Mol. Graph.* **1994**, *12*, 84−89.

(3) Markwick, P. R.; McCammon, J. A. Studying functional dynamics in bio-molecules using accelerated molecular dynamics. *Phys. Chem. Chem. Phys.* **2011**, *13*, 20053−20065.

(4) Shaw, D. E.; Dror, R. O.; Salmon, J. K.; Grossman, J. P.; Mackenzie, K. M.; Bank, J. A.; Young, C.; Deneroff, M. M.; Batson, B.; Bowers, K. J. et al. Millisecond-scale molecular dynamics simulations on Anton. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis (SC '09)*, Portland, OR, November 2009; ACM Press, 2009; pp 1−11, Article No. 39, .

(5) Lindorff-Larsen, K.; Maragakis, P.; Piana, S.; Shaw, D. E. Picosecond to Millisecond Structural Dynamics in Human Ubiquitin. *J. Phys. Chem. B* **2016**, *120*, 8313−8320.

(6) Jaffrelot Inizan, T.; Célerse, F.; Adjoua, O.; El Ahdab, D.; Jolly, L.-H.; Liu, C.; Ren, P.; Montes, M.; Lagarde, N.; Lagardère, L.; Monmarché, P.; Piquemal, J.-P. High-resolution mining of the SARS-CoV-2 main protease conformational space: supercomputer-driven unsupervised adaptive sampling. *Chem. Sci.* **2021**, *12*, 4889−4907.

(7) Cho, K.; Joannopoulos, J. Ergodicity and dynamical properties of constant-temperature molecular dynamics. *Phys. Rev. A* **1992**, *45*, 7089.

(8) Hamelberg, D.; Mongan, J.; McCammon, J. A. Accelerated molecular dynamics: a promising and efficient simulation method for biomolecules. *J. Chem. Phys.* **2004**, *120*, 11919−11929.

(9) Voter, A. F. Hyperdynamics: Accelerated molecular dynamics of infrequent events. *Phys. Rev. Lett.* **1997**, *78*, 3908.

(10) Pierce, L. C. T.; Salomon-Ferrer, R.; de Oliveira, C. A.; McCammon, J. A.; Walker, R. C. Routine access to millisecond time scale events with accelerated molecular dynamics. *J. Chem. Theory. Comput.* **2012**, *8*, 2997−3002.

(11) Miao, Y.; Feixas, F.; Eun, C.; McCammon, J. A. Accelerated molecular dynamics simulations of protein folding. *J. Comput. Chem.* **2015**, *36*, 1536−1549.

(12) Lagardère, L.; Jolly, L.-H.; Lipparini, F.; Aviat, F.; Stamm, B.; Jing, Z. F.; Harger, M.; Torabifard, H.; Cisneros, G. A.; Schnieders, M. J.; Gresh, N.; Maday, Y.; Ren, P. Y.; Ponder, J. W.; Piquemal, J.-P.; et al. Tinker-HP: A massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields. *Chem. Sci.* **2018**, *9*, 956−972.

(13) Götz, A. W.; Williamson, M. J.; Xu, D.; Poole, D.; Le Grand, S.; Walker, R. C. Routine Microsecond Molecular Dynamics Simulations with AMBER on GPUs. 1. Generalized Born. *J. Chem. Theory Comput.* **2012**, *8*, 1542−1555.

(14) Páll, S.; Zhmurov, A.; Bauer, P.; Abraham, M.; Lundborg, M.; Gray, A.; Hess, B.; Lindahl, E. Heterogeneous parallelization and acceleration of molecular dynamics simulations in GROMACS. *J. Chem. Phys.* **2020**, *153*, 134110.

(15) Adjoua, O.; Lagardere, L.; Jolly, L.-H.; Durocher, A.; Very, T.; Dupays, I.; Wang, Z.; Inizan, T. J.; Celerse, F.; Ren, P.; Ponder, J. W.; Piquemal, J.-P.; et al. Tinker-HP: Accelerating Molecular Dynamics Simulations of Large Complex Systems with Advanced Point Dipole Polarizable Force Fields Using GPUs and Multi-GPU Systems. *J. Chem. Theory. Comput.* **2021**, *17*, 2034−2053.

(16) Tuckerman, M. E.; Berne, B. J.; Rossi, A. Molecular dynamics algorithm for multiple time scales: Systems with disparate masses. *J. Chem. Phys.* **1991**, *94*, 1465−1469.

(17) Lagardère, L.; Aviat, F.; Piquemal, J.-P. Pushing the limits of multiple-time-step strategies for polarizable point dipole molecular dynamics. *J. Phys. Chem. Lett.* **2019**, *10*, 2593−2599.

(18) Fiorin, G.; Klein, M. L.; Hénin, J. Using collective variables to drive molecular dynamics simulations. *Mol. Phys.* **2013**, *111*, 3345−3362.

(19) Tribello, G. A.; Bonomi, M.; Branduardi, D.; Camilloni, C.; Bussi, G. PLUMED 2: New feathers for an old bird. *Comput. Phys. Commun.* **2014**, *185*, 604−613.

(20) Brotzakis, Z. F.; Limongelli, V.; Parrinello, M. Accelerating the calculation of protein−ligand binding free energy and residence times using dynamically optimized collective variables. *J. Chem. Theory. Comput.* **2019**, *15*, 743−750.

(21) Kabelka, I.; Brozek, R.; Vácha, R. Selecting Collective Variables and Free-Energy Methods for Peptide Translocation across Membranes. *J. Chem. Inf. Model.* **2021**, *61*, 819−830.

(22) Hovan, L.; Comitani, F.; Gervasio, F. L. Defining an optimal metric for the path collective variables. *J. Chem. Theory. Comput.* **2019**, *15*, 25−32.

(23) Bonati, L.; Rizzi, V.; Parrinello, M. Data-driven collective variables for enhanced sampling. *J. Phys. Chem. Lett.* **2020**, *11*, 2998−3004.

(24) Hashemian, B.; Millán, D.; Arroyo, M. Modeling and enhanced sampling of molecular systems with smooth and nonlinear data-driven collective variables. *J. Chem. Phys.* **2013**, *139*, 214101.

(25) Chen, P.-Y.; Tuckerman, M. E. Molecular dynamics based enhanced sampling of collective variables with very large time steps. *J. Chem. Phys.* **2018**, *148*, 024106.

(26) Henin, J.; Fiorin, G.; Chipot, C.; Klein, M. L. Exploring multidimensional free energy landscapes using time-dependent biases on collective variables. *J. Chem. Theory. Comput.* **2010**, *6*, 35−47.
(27) Gardner, J. M.; Abrams, C. F. Energetics of flap opening in HIV-1 protease: string method calculations. *J. Phys. Chem. B* **2019**, *123*, 9584−9591.
(28) Célerse, F.; Lagardère, L.; Derat, E.; Piquemal, J.-P. Massively parallel implementation of Steered Molecular Dynamics in Tinker-HP: comparisons of polarizable and non-polarizable simulations of realistic systems. *J. Chem. Theory. Comput.* **2019**, *15*, 3694−3709.
(29) Rodriguez, A.; d'Errico, M.; Facco, E.; Laio, A. Computing the free energy without collective variables. *J. Chem. Theory. Comput.* **2018**, *14*, 1206−1215.
(30) Miao, Y.; Feher, V. A.; McCammon, J. A. Gaussian accelerated molecular dynamics: Unconstrained enhanced sampling and free energy calculation. *J. Chem. Theory. Comput.* **2015**, *11*, 3584−3595.
(31) Melcr, J.; Piquemal, J.-P. Accurate Biomolecular Simulations Account for Electronic Polarization. *Front. Mol. Biosci.* **2019**, *6*, 143.
(32) Shi, Y.; Ren, P.; Schnieders, M.; Piquemal, J.-P. *Reviews in Computational Chemistry*, Vol. *28*; John Wiley & Sons, Ltd., 2015; Chapter 2, pp 51−86; DOI: 10.1002/9781118889886.ch2.
(33) Jing, Z.; Liu, C.; Cheng, S. Y.; Qi, R.; Walker, B. D.; Piquemal, J.-P.; Ren, P. Polarizable Force Fields for Biomolecular Simulations: Recent Advances and Applications. *Ann. Rev. Biophys.* **2019**, *48*, 371−394.
(34) Gresh, N.; Cisneros, G. A.; Darden, T. A.; Piquemal, J.-P. Anisotropic, Polarizable Molecular Mechanics Studies of Inter- and Intramolecular Interactions and Ligand-Macromolecule Complexes. A Bottom-Up Strategy. *J. Chem. Theory Comput.* **2007**, *3*, 1960−1986.
(35) Ren, P.; Ponder, J. W. Polarizable atomic multipole water model for molecular mechanics simulation. *J. Phys. Chem. B* **2003**, *107*, 5933−5947.
(36) Ponder, J. W.; Wu, C.; Ren, P.; Pande, V. S.; Chodera, J. D.; Schnieders, M. J.; Haque, I.; Mobley, D. L.; Lambrecht, D. S.; DiStasio, R. A.; Head-Gordon, M.; Clark, G. N. I.; Johnson, M. E.; Head-Gordon, T.; et al. Current status of the AMOEBA polarizable force field. *J. Phys. Chem. B* **2010**, *114*, 2549−2564.
(37) Torrie, G. M.; Valleau, J. P. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.* **1977**, *23*, 187−199.
(38) Oshima, H.; Re, S.; Sugita, Y. Replica-exchange umbrella sampling combined with gaussian accelerated molecular dynamics for free-energy calculation of biomolecules. *J. Chem. Theory. Comput.* **2019**, *15*, 5199−5208.
(39) Miao, Y.; Bhattarai, A.; Wang, J. Ligand Gaussian accelerated molecular dynamics (LiGaMD): Characterization of ligand binding thermodynamics and kinetics. *J. Chem. Theory. Comput.* **2020**, *16*, 5526−5547.
(40) Wang, J.; Miao, Y. Peptide Gaussian accelerated molecular dynamics (Pep-GaMD): Enhanced sampling and free energy and kinetics calculations of peptide binding. *J. Chem. Phys.* **2020**, *153*, 154109.
(41) Miao, Y.; McCammon, J. A. Gaussian Accelerated Molecular Dynamics: Theory, Implementation, and Applications. In *Annual Reports in Computational Chemistry*, Vol. *13*; Elsevier, 2017; pp 231−278, .
(42) Wang, J.; Arantes, P. R.; Bhattarai, A.; Hsu, R. V.; Pawnikar, S.; Huang, Y.-m. M.; Palermo, G.; Miao, Y. Gaussian accelerated molecular dynamics: Principles and applications. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2021**, *11*, e1521.
(43) Wang, Y.-T.; Chan, Y.-H. Understanding the molecular basis of agonist/antagonist mechanism of human mu opioid receptor through gaussian accelerated molecular dynamics method. *Sci. Rep.* **2017**, *7*, 7828.
(44) Palermo, G. Structure and dynamics of the CRISPR−Cas9 catalytic complex. *J. Chem. Inf. Model.* **2019**, *59*, 2394−2406.
(45) de Courcy, B.; Piquemal, J.-P.; Garbay, C.; Gresh, N. Polarizable water molecules in ligand- macromolecule recognition. Impact on the relative affinities of competing pyrrolopyrimidine inhibitors for FAK kinase. *J. Am. Chem. Soc.* **2010**, *132*, 3312−3320.
(46) El Ahdab, D.; Lagardère, L.; Inizan, T. J.; Célerse, F.; Liu, C.; Adjoua, O.; Jolly, L.-H.; Gresh, N.; Hobaika, Z.; Ren, P.; Maroun, R. G.; Piquemal, J.-P. Interfacial Water Many-Body Effects Drive Structural Dynamics and Allosteric Interactions in SARS-CoV-2 Main Protease Dimerization Interface. *J. Phys. Chem. Lett.* **2021**, *12*, 6218−6226.
(47) Leimkuhler, B.; Matthews, C. Robust and efficient configurational molecular sampling via Langevin dynamics. *J. Chem. Phys.* **2013**, *138*, 174102.
(48) Mark, P.; Nilsson, L. Structure and dynamics of the TIP3P, SPC, and SPC/E water models at 298 K. *J. Phys. Chem. A* **2001**, *105*, 9954−9960.
(49) MacKerell, A. D.; Bashford, D.; Bellott, M.; Dunbrack, R. L.; Evanseck, J. D.; Field, M. J.; Fischer, S.; Gao, J.; Guo, H.; Ha, S.; Joseph-McCarthy, D.; Kuchnir, L.; Kuczera, K.; Lau, F. T. K.; Mattos, C.; Michnick, S.; Ngo, T.; Nguyen, D. T.; Prodhom, B.; Reiher, W. E.; Roux, B.; Schlenkrich, M.; Smith, J. C.; Stote, R.; Straub, J.; Watanabe, M.; Wiorkiewicz-Kuczera, J.; Yin, D.; Karplus, M.; et al. All-atom empirical potential for molecular modeling and dynamics studies of proteins. *J. Phys. Chem. B* **1998**, *102*, 3586−3616.
(50) Shi, Y.; Xia, Z.; Zhang, J.; Best, R.; Wu, C.; Ponder, J. W.; Ren, P. Polarizable atomic multipole-based AMOEBA force field for proteins. *J. Chem. Theory. Comput.* **2013**, *9*, 4046−4063.
(51) Zhang, C.; Lu, C.; Jing, Z.; Wu, C.; Piquemal, J.-P.; Ponder, J. W.; Ren, P. AMOEBA polarizable atomic multipole force field for nucleic acids. *J. Chem. Theory. Comput.* **2018**, *14*, 2084−2108.
(52) Bussi, G.; Donadio, D.; Parrinello, M. Canonical sampling through velocity rescaling. *J. Chem. Phys.* **2007**, *126*, 014101.
(53) Chow, K.-H.; Ferguson, D. M. Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Comput. Phys. Commun.* **1995**, *91*, 283−289.
(54) Swope, W. C.; Andersen, H. C.; Berens, P. H.; Wilson, K. R. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *J. Chem. Phys.* **1982**, *76*, 637−649.
(55) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A smooth particle mesh Ewald method. *J. Chem. Phys.* **1995**, *103*, 8577−8593.
(56) You, W.; Tang, Z.; Chang, C.-e. A. Potential mean force from umbrella sampling simulations: What can we learn and what is missed? *J. Chem. Theory. Comput.* **2019**, *15*, 2433−2443.
(57) Baştuğ, T.; Chen, P.-C.; Patra, S. M.; Kuyucak, S. Potential of mean force calculations of ligand binding to ion channels from Jarzynski's equality and umbrella sampling. *J. Chem. Phys.* **2008**, *128*, 155104.
(58) Mills, M.; Andricioaei, I. An experimentally guided umbrella sampling protocol for biomolecules. *J. Chem. Phys.* **2008**, *129*, 114101.
(59) Bayas, M. V.; Kearney, A.; Avramovic, A.; Van Der Merwe, P. A.; Leckband, D. E. Impact of salt bridges on the equilibrium binding and adhesion of human CD2 and CD58. *J. Biol. Chem.* **2007**, *282*, 5589−5596.
(60) Debiec, K. T.; Gronenborn, A. M.; Chong, L. T. Evaluating the strength of salt bridges: a comparison of current biomolecular force fields. *J. Phys. Chem. B* **2014**, *118*, 6561−6569.
(61) Shirts, M. R.; Chodera, J. D. Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.* **2008**, *129*, 124105.
(62) Ding, X.; Vilseck, J. Z.; Hayes, R. L.; Brooks, C. L. Gibbs Sampler-Based -Dynamics and Rao−Blackwell Estimator for Alchemical Free Energy Calculation. *J. Chem. Theory Comput.* **2017**, *13*, 2501−2510.
(63) Ding, X.; Vilseck, J. Z.; Brooks, C. L. Fast Solver for Large Scale Multistate Bennett Acceptance Ratio Equations. *J. Chem. Theory. Comput.* **2019**, *15*, 799−802.

Conclusion

The combination of these sampling techniques, e.g ASUS-GaMD, with GPUs has led to a significant reduction in the time required to evaluate free energy profiles using PFFs. We provided evidence that on a large biomolecular system that the ASUS-GaMD is 1.4 times faster than classical US. With the inherent parallelism inherited from adaptive sampling, the PMF evaluation can be completed in only one-fifth of the simulation time, resulting in a speedup of 7. Moreover, considering that convergence was already achieved with a 25ns per window setup, this factor increases to 14. As a result, ASUS-GaMD is capable of reducing computations that would have taken months to just a matter of days. Furthermore, this work facilitates the utilization of various combined approaches, offering access to GPU-accelerated GaMD-adaptive sampling simulations, which will be instrumental in expanding conformational space studies of proteins.

Conclusion

Throughout this thesis, Machine Learning (ML) models have been utilized at various levels to mitigate some limitations of large-scale Molecular Dynamics (MD). While physically-motivated empirical Force Fields (FFs) offer computational advantages such as their low cost, they may not accurately capture quantum mechanical effects. Additionally, an efficient sampling of the conformational space of molecules remains a challenge, particularly in biomolecular simulations where the timescale of interest exceeds the accessible timescale in MD. Furthermore, analyzing and interpreting the high-dimensional nature of large simulation trajectories, especially in the case of biomolecular simulations where Collective Variables (CVs) are often unknown, remains an open question.

In recent years, extensive efforts have been made to incorporate ML potentials (MLPs) into FFs, as ML holds the promise of bridging the accuracy and generality gap between FFs and ab-initio models. While FFs and ab-initio models are based on physical intuition and grounded in physics, making them easily transferable, FFs rely on approximate formulas and lack accuracy, while ab-initio models are computationally expensive and unsuitable for studying large systems, such as those found in biology. On the other hand, most MLP models tend to overlook long-range effects critical for accurately simulating condensed-phase systems and describing the structure of large protein or DNA structures. Moreover, they often neglect Nuclear Quantum Effects (NQEs). With these considerations in mind, several ML-based physics-aware models have been introduced in this thesis.

The first model, known as DNN-MBD, incorporates ML to circumvent the computationally expensive quantum mechanical calculations involved in electron density partitioning. The ML model is trained on local atomic properties, specifically Atom-In-Molecule volumes at the MBISA level. This approach broadens its applicability beyond electronic structure theory to methodologies such as FFs and neural networks, making it highly valuable for simulations of large and complex systems. Evaluation of the DNN-MBD model on the well-known S66x8 benchmark set, coupled with common PBE/PBE0 density functionals, shows comparable performance to CCSD(T)/CBS, with an error of only 0.25 kcal/mol. These advancements pave the way for generating extensive and highly accurate datasets for future machine learning models, offering new avenues for research and development in the field.

Considerable focus has also been devoted in this thesis to combining MLPs with Polarizable FFs (PFFs). To achieve this, it was necessary to develop a highly parallel multi-GPU ML platform and integrate it into an existing MD software. Here we developed Deep-HP which is part of the Tinker-HP package. This platform empowers users to combine MLP models with FFs through Tinker-HP, enabling simulations of millions of atoms and routine production simulations of large biomolecular systems with advanced neural network models. The platform's capabilities were demonstrated by simulating large biologically-relevant systems using state-of-the-art MLP models on hundreds of GPUs. This platform led to the development of an hybrid model that combines the ANI-2X MLP and AMOEBA PFFs, incorporating physically-motivated long-range effects through AMOEBA. To

enhance the hybrid model's capabilities, sophisticated strategies based on multi-time-step integrators significantly accelerated simulations by a factor of 20. It was demonstrated that this setup allows for alchemical free energy computations. Evaluations of the hybrid model's accuracy included assessing solvation free energies of 70 molecules in various solvents and binding free energies of 14 challenging host-guest systems from the SAMPL host-guest binding competitions. The hybrid model was able to perform better than AMOEBA reference data, achieving an accuracy within the range of chemical accuracy, with an error of 0.94 kcal/mol compared to AMOEBA's 1.81 kcal/mol on the SAMPL challenge.

To tackle the challenge of developing a PFFs model based on machine learning for biomolecular simulations, we are developing the Q-AMOEBA-NN model. This model will leverage quantum-accurate long-range interactions through the AMOEBA PFF while employing ML model short-range interactions. In order to maintain stability and prevent reactivity, the bond-stretching interactions are described by AMOEBA. Additionally, an additional ML model was employed to refine the vdW parameters, eliminating the additional NQEs that were present in the original parametrization of AMOEBA. The development of the Q-AMOEBA-NN model is made possible through the parametrization of a large database containing millions of conformations, including dipeptides, dimers, water clusters, and solvated ions, using the AMOEBA force field. The performance of the model will be assess on various MD properties and free energy computations and hoping it will opens up new avenues for exploring complex biomolecular phenomena at the quantum-level.

Furthermore, in addition to the accuracy of the potential model, efficient sampling of the conformational space is crucial. This thesis has focused on the development of data-driven enhanced sampling techniques that are CVs-free. One of them is the Adaptive Sampling (AS), which involves iterations of parallel independent molecular dynamics (MD) replicas. The AS algorithm selects the initial structure of each MD replica based on a probability inversely proportional to its density in a low-dimensional space. This low-dimensional space can be obtained using various dimensional reduction algorithms found in ML from component analysis methods to autoencoders and variational autoencoders. The effectiveness of the AS algorithm was demonstrated in sampling the conformational space of the SARS-CoV-2 M^{pro}. It enabled the generation of more than 50 μs of all-atom MD simulations using the AMOEBA FF, which represents the longest simulation conducted with PFF to date.

To further enhance the sampling efficiency, the adaptive sampling algorithm was combined with a novel generalized accelerated molecular dynamics (GaMD) method specifically designed for multi-time-step integrators and PFFs. Additionally, the technique of Umbrella Sampling was also incorporated. The combination of these sampling techniques resulted in a significant reduction in the computational time required to evaluate free energy profiles. Notably, the ASUS-GaMD coupling achieved a speed factor of 15 in converging the free energy profile, demonstrating its effectiveness in enhancing sampling efficiency.

In our previous discussion, we explored how ML can enhance model accuracy and accelerate the exploration of the conformational space of a molecule, enabling the production of simulation trajectories ranging from μs to milliseconds. However, an equally important aspect to address is how ML can effectively analyze such vast amounts of production data. In this thesis, we employed cutting-edge clustering algorithms such as HDBSCAN and OPTICS to extract biological structure patterns from

clusters. Through this analysis, previously unknown behaviors were unveiled, including long-range cooperative conformational changes in the M^{pro} protein, known as allosteric interactions, as well as cryptic pocket shielding which were later confirmed by experiments. Furthermore, this analysis revealed hidden cryptic pockets that played a significant role in the design of novel inhibitors and helped decipher the crucial stability mechanism of the M^{pro} protein at physiological pH.

Additionally, a deep learning-driven Hidden Markov State analysis was employed to investigate the binding modes of a set of covalent M^{pro} inhibitors, which were designed based on the 50 μs of simulations obtained using the AS algorithm. This analysis, in conjunction with the k-means clustering algorithm, revealed the simultaneous presence of multiple binding modes within a cryptic pocket, aligning closely with experimental observations. This finding provides strong agreement between the computational analysis and real-world experiences.

The extensive research conducted in this thesis thus far exemplifies the diverse applications of ML in addressing global challenges. It encompasses endeavors to improve the accuracy of force fields, enhance sampling techniques, and gain valuable insights into ligand binding modes. The primary objective of this thesis is to pave new pathways in molecular modeling by harnessing the power of ML models, statistical mechanics, and parallel GPU computing. As a result, it offers novel approaches for studying a wider range of systems with enhanced precision and efficiency. This is accomplished through the utilization of ML-driven GaMD-AS strategies, in conjunction with efficient multi-GPU platforms like Deep-HP and Quantum-HP, as well as the utilization of accurate hybrid ML PFFs models. Looking forward, the ongoing development and future refinement of the Q-AMOEBA-NN model, combined with larger and more precise datasets, along with the potential integration of DNN-MBD, hold tremendous potential for realizing quantum-accurate simulations of billion-atom systems on the millisecond timescale.

Bibliography

[1]M. Karplus, R. N. Porter, and R. D. Sharma, “Dynamics of reactive collisions: the h +h2 exchange reaction”, The Journal of Chemical Physics **40**, 2033–2034 (1964) (cit. on pp. v, 3).

[2]L. Verlet, “Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules”, Phys. Rev. **159**, 98–103 (1967) (cit. on pp. v, 3, 15).

[3]A. Rahman, “Correlations in the motion of atoms in liquid argon”, Phys. Rev. **136**, A405–A411 (1964) (cit. on pp. v, 3).

[4]Y. Shi, P. Ren, M. Schnieders, and J.-P. Piquemal, “Polarizable force fields for biomolecular modeling”, in *Reviews in computational chemistry volume 28* (John Wiley and Sons, Ltd, 2015) Chap. 2, pp. 51–86 (cit. on pp. v, 3).

[5]Z. Jing, C. Liu, S. Y. Cheng, et al., “Polarizable force fields for biomolecular simulations: recent advances and applications”, Annual Review of Biophysics **48**, 371–394 (2019) (cit. on pp. v, 3).

[6]A. Paszke, S. Gross, F. Massa, et al., “Pytorch: an imperative style, high-performance deep learning library”, in *Advances in neural information processing systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, et al. (Curran Associates, Inc., 2019), pp. 8024–8035 (cit. on pp. vi, 4).

[7]Martín Abadi, Ashish Agarwal, Paul Barham, et al., *TensorFlow: large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015 (cit. on pp. vi, 4).

[8]F. Pedregosa, G. Varoquaux, A. Gramfort, et al., “Scikit-learn: machine learning in Python”, Journal of Machine Learning Research **12**, 2825–2830 (2011) (cit. on pp. vi, 4).

[9]F. Chollet et al., *Keras*, 2015 (cit. on pp. vi, 4).

[10]A. D. MacKerell Jr., B. Brooks, C. L. Brooks III, et al., “Charmm: the energy function and its parameterization”, in *Encyclopedia of computational chemistry* (Wiley and Sons, 2002) (cit. on pp. vi, 4, 19).

[11]M. J. Abraham, T. Murtola, R. Schulz, et al., “Gromacs: high performance molecular simulations through multi-level parallelism from laptops to supercomputers”, SoftwareX **1-2**, 19–25 (2015) (cit. on pp. vi, 4, 15).

[12]L. Lagardère, L.-H. Jolly, F. Lipparini, et al., “Tinker-hp: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields”, Chem. Sci. **9**, 956–972 (2018) (cit. on pp. vi, 4, 28).

[13]O. Adjoua, L. Lagardère, L.-H. Jolly, et al., “Tinker-hp: accelerating molecular dynamics simulations of large complex systems with advanced point dipole polarizable force fields using gpus and multi-gpu systems”, Journal of Chemical Theory and Computation **17**, PMID: 33755446, 2034–2053 (2021) (cit. on pp. vi, 4, 172).

[14]P. P. Poier, L. Lagardère, and J.-P. Piquemal, “O(n) stochastic evaluation of many-body van der waals energies in large complex systems”, Journal of Chemical Theory and Computation **18**, PMID: 35133157, 1633–1645 (2022) (cit. on pp. vii, 4, 13, 85).

[15]P. P. Poier, T. Jaffrelot Inizan, O. Adjoua, L. Lagardère, and J.-P. Piquemal, "Accurate deep learning-aided density-free strategy for many-body dispersion-corrected density functional theory", The Journal of Physical Chemistry Letters **13**, PMID: 35544748, 4381–4388 (2022) (cit. on pp. ix, 55, 85).

[16]L. Goerigk, H. Kruse, and S. Grimme, "Benchmarking density functional methods against the s66 and s66x8 datasets for non-covalent interactions", ChemPhysChem **12**, 3421–3433 (2011) (cit. on p. x).

[17]A. Ambrosetti, A. M. Reilly, R. A. DiStasio, and A. Tkatchenko, "Long-range correlation energy calculated from coupled atomic response functions", The Journal of Chemical Physics **140**, 18A508 (2014) (cit. on pp. x, 13, 55).

[18]T. Gould, S. Lebègue, J. G. Ángyán, and T. Bučko, "A fractionally ionic approach to polarizability and van der waals many-body dispersion calculations", Journal of Chemical Theory and Computation **12**, 5920–5930 (2016) (cit. on p. x).

[19]T. W. Ko, J. A. Finkler, S. Goedecker, and J. Behler, "A fourth-generation high-dimensional neural network potential with accurate electrostatics including non-local charge transfer", Nature Communications **12**, 398 (2021) (cit. on pp. 4, 67).

[20]J. Behler and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces", Phys. Rev. Lett. **98**, 146401 (2007) (cit. on pp. 4, 34, 35).

[21]P. A. M. Dirac, *The principles of quantum mechanics* (Clarendon Press, 1930) (cit. on p. 7).

[22]B. Diu, C. cohen-tannoudji, and F. Laloe, "Quantum mechanics:" Fundamentals of Physics II (2020) (cit. on p. 7).

[23]L. de Broglie, "Recherches sur la théorie des Quanta", Theses (Migration - université en cours d'affectation, Nov. 1924) (cit. on p. 7).

[24]M. Born and R. Oppenheimer, "Zur quantentheorie der molekeln", Annalen der Physik **389**, 457–484 (1927) (cit. on p. 8).

[25]D. R. Hartree, "The wave mechanics of an atom with a non-coulomb central field. part ii. some results and discussion", Mathematical Proceedings of the Cambridge Philosophical Society **24**, 111–132 (1928) (cit. on p. 9).

[26]C. Møller and M. S. Plesset, "Note on an approximation treatment for many-electron systems", Phys. Rev. **46**, 618–622 (1934) (cit. on p. 9).

[27]Y. Yao, E. Giner, J. Li, J. Toulouse, and C. J. Umrigar, "Almost exact energies for the Gaussian-2 set with the semistochastic heat-bath configuration interaction method", The Journal of Chemical Physics **153**, 124117, 10.1063/5.0018577 (2020) (cit. on p. 10).

[28]E. Giner, R. Assaraf, and J. Toulouse, *Quantum monte carlo with reoptimized perturbatively selected configuration-interaction wave functions*, 2016 (cit. on p. 10).

[29]E. Giner, A. Scemama, and M. Caffarel, "Using perturbatively selected configuration interaction in quantum monte carlo calculations", Canadian Journal of Chemistry **91**, 879–885 (2013) (cit. on p. 10).

[30]A. A. Holmes, N. M. Tubman, and C. J. Umrigar, "Heat-bath configuration interaction: an efficient selected configuration interaction algorithm inspired by heat-bath sampling", Journal of Chemical Theory and Computation **12**, PMID: 27428771, 3674–3680 (2016) (cit. on p. 10).

[31]S. Evangelisti, J.-P. Daudey, and J.-P. Malrieu, "Convergence of an improved cipsi algorithm", Chemical Physics **75**, 91–102 (1983) (cit. on p. 10).

[32] Y. Guo, C. Riplinger, U. Becker, et al., "Communication: an improved linear scaling perturbative triples correction for the domain based local pair-natural orbital based singles and doubles coupled cluster method [dlpno-ccsd(t)]", The Journal of Chemical Physics **148**, 011101 (2018) (cit. on p. 11).

[33] P. Hohenberg and W. Kohn, "Inhomogeneous electron gas", Phys. Rev. **136**, B864–B871 (1964) (cit. on p. 11).

[34] W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects", Phys. Rev. **140**, A1133–A1138 (1965) (cit. on pp. 11, 55).

[35] J. P. Perdew, K. Burke, and M. Ernzerhof, "Generalized gradient approximation made simple", Phys. Rev. Lett. **77**, 3865–3868 (1996) (cit. on p. 12).

[36] A. D. Becke, "A new mixing of Hartree–Fock and local density-functional theories", The Journal of Chemical Physics **98**, 1372–1377 (1993) (cit. on p. 12).

[37] A. D. Becke, "Density-functional thermochemistry. III. The role of exact exchange", The Journal of Chemical Physics **98**, 5648–5652 (1993) (cit. on p. 12).

[38] C. Adamo and V. Barone, "Toward reliable density functional methods without adjustable parameters: The PBE0 model", The Journal of Chemical Physics **110**, 6158–6170 (1999) (cit. on p. 12).

[39] S. Grimme, "Accurate description of van der waals complexes by density functional theory including empirical corrections", Journal of Computational Chemistry **25**, 1463–1473 (2004) (cit. on p. 13).

[40] S. Grimme, "Semiempirical gga-type density functional constructed with a long-range dispersion correction", Journal of Computational Chemistry **27**, 1787–1799 (2006) (cit. on p. 13).

[41] S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, "A consistent and accurate ab initio parametrization of density functional dispersion correction (dft-d) for the 94 elements h-pu", The Journal of Chemical Physics **132**, 154104 (2010) (cit. on p. 13).

[42] A. Tkatchenko and M. Scheffler, "Accurate molecular van der waals interactions from ground-state electron density and free-atom reference data", Phys. Rev. Lett. **102**, 073005 (2009) (cit. on pp. 13, 55).

[43] F. L. Hirshfeld, "Bonded-atom fragments for describing molecular charge densities", Theoretica chimica acta **44**, 129–138 (1977) (cit. on pp. 13, 55).

[44] S. Ubaru, J. Chen, and Y. Saad, "Fast estimation of $tr(f(a))$ via stochastic lanczos quadrature", SIAM Journal on Matrix Analysis and Applications **38**, 1075–1099 (2017) (cit. on p. 13).

[45] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson, "A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: application to small water clusters", The Journal of Chemical Physics **76**, 637–649 (1982) (cit. on p. 15).

[46] S. Nosé, "A unified formulation of the constant temperature molecular dynamics methods", The Journal of Chemical Physics **81**, 511–519 (1984) (cit. on p. 15).

[47] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, "Molecular dynamics with coupling to an external bath", The Journal of Chemical Physics **81**, 3684–3690 (1984) (cit. on p. 15).

[48] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, "Molecular dynamics with coupling to an external bath", The Journal of Chemical Physics **81**, 3684–3690 (1984) (cit. on p. 15).

[49] G. Bussi, D. Donadio, and M. Parrinello, "Canonical sampling through velocity rescaling", The Journal of Chemical Physics **126**, 014101, 10.1063/1.2408420 (2007) (cit. on p. 15).

[50]D. Frenkel and B. Smit, "Chapter 1 - introduction", in *Understanding molecular simulation (second edition)*, edited by D. Frenkel and B. Smit, Second Edition (Academic Press, San Diego, 2002), pp. 1–6 (cit. on pp. 15, 42).

[51]G. J. Martyna, M. E. Tuckerman, D. J. Tobias, and M. L. Klein, "Explicit reversible integrators for extended systems dynamics", Molecular Physics **87**, 1117–1157 (1996) (cit. on p. 15).

[52]W. L. Jorgensen and J. Tirado-Rives, "The opls [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin", Journal of the American Chemical Society **110**, PMID: 27557051, 1657–1666 (1988) (cit. on pp. 15, 18).

[53]D. A. Case, T. E. Cheatham 3rd, T. Darden, et al., "The amber biomolecular simulation programs", en, J Comput Chem **26**, 1668–1688 (2005) (cit. on pp. 15, 18).

[54]B. R. Brooks, C. L. Brooks III, A. D. Mackerell Jr., et al., "Charmm: the biomolecular simulation program", Journal of Computational Chemistry **30**, 1545–1614 (2009) (cit. on pp. 15, 18).

[55]C. Zhang, C. Lu, Z. Jing, et al., "Amoeba polarizable atomic multipole force field for nucleic acids", Journal of Chemical Theory and Computation **14**, 2084–2108 (2018) (cit. on p. 15).

[56]J. W. Ponder, C. Wu, P. Ren, et al., "Current status of the amoeba polarizable force field", The Journal of Physical Chemistry B **114**, 2549–2564 (2010) (cit. on p. 15).

[57]C. Liu, J.-P. Piquemal, and P. Ren, "Amoeba+ classical potential for modeling molecular interactions", Journal of Chemical Theory and Computation **15**, 4122–4139 (2019) (cit. on p. 15).

[58]N. L. Allinger, Y. H. Yuh, and J. H. Lii, "Molecular mechanics. the mm3 force field for hydrocarbons. 1", Journal of the American Chemical Society **111**, 8551–8566 (1989) (cit. on p. 17).

[59]A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. I. Goddard, and W. M. Skiff, "Uff, a full periodic table force field for molecular mechanics and molecular dynamics simulations", Journal of the American Chemical Society **114**, 10024–10035 (1992) (cit. on p. 17).

[60]S. Lifson and A. Warshel, "Consistent Force Field for Calculations of Conformations, Vibrational Spectra, and Enthalpies of Cycloalkane and n-Alkane Molecules", The Journal of Chemical Physics **49**, 5116–5129 (2003) (cit. on p. 17).

[61]J. Huang, S. Rauscher, G. Nawrocki, et al., "Charmm36m: an improved force field for folded and intrinsically disordered proteins", Nature Methods **14**, 71–73 (2017) (cit. on pp. 18, 19).

[62]A. Stone and M. Alderton, "Distributed multipole analysis", Molecular Physics **56**, 1047–1064 (1985) (cit. on p. 19).

[63]A. Stone, *The Theory of Intermolecular Forces* (Oxford University Press, Jan. 2013) (cit. on p. 19).

[64]B. Thole, "Molecular polarizabilities calculated with a modified dipole interaction", Chemical Physics **59**, 341–350 (1981) (cit. on p. 19).

[65]F. Lipparini, L. Lagardère, C. Raynaud, et al., "Polarizable molecular dynamics in a polarizable continuum solvent", Journal of Chemical Theory and Computation **11**, 623–634 (2015) (cit. on p. 20).

[66]D. Loco, L. Lagardère, S. Caprasecca, et al., "Hybrid qm/mm molecular dynamics with amoeba polarizable embedding", Journal of Chemical Theory and Computation **13**, PMID: 28759205, 4025–4033 (2017) (cit. on p. 20).

[67]D. Bedrov, J.-P. Piquemal, O. Borodin, et al., "Molecular dynamics simulations of ionic liquids and electrolytes using polarizable force fields", Chemical Reviews **119**, PMID: 31141351, 7940–7995 (2019) (cit. on p. 20).

[68]P. E. M. Lopes, B. Roux, and A. D. Mackerell Jr, "Molecular modeling and dynamics studies with explicit inclusion of electronic polarizability. theory and applications", en, Theor Chem Acc **124**, 11–28 (2009) (cit. on p. 20).

[69]P. Ren and J. W. Ponder, "Polarizable atomic multipole water model for molecular mechanics simulation", The Journal of Physical Chemistry B **107**, 5933–5947 (2003) (cit. on p. 20).

[70]J. C. Wu, G. Chattree, and P. Ren, "Automation of AMOEBA polarizable force field parameterization for small molecules", en, Theor Chem Acc **131**, 1138 (2012) (cit. on p. 21).

[71]B. Walker, C. Liu, E. Wait, and P. Ren, "Automation of amoeba polarizable force field for small molecules: poltype 2", Journal of Computational Chemistry **43**, 1530–1542 (2022) (cit. on pp. 21, 23).

[72]N. M. O'Boyle, M. Banck, C. A. James, et al., "Open babel: an open chemical toolbox", Journal of Cheminformatics **3**, 33 (2011) (cit. on p. 22).

[73]L.-P. Wang, T. Head-Gordon, J. W. Ponder, et al., "Systematic improvement of a classical molecular model of water", The Journal of Physical Chemistry B **117**, PMID: 23750713, 9956–9972 (2013) (cit. on p. 22).

[74]N. Mauger, T. Plé, L. Lagardère, et al., "Nuclear quantum effects in liquid water at near classical computational cost using the adaptive quantum thermal bath", The Journal of Physical Chemistry Letters **12**, PMID: 34427440, 8285–8291 (2021) (cit. on pp. 22, 25).

[75]N. Mauger, T. Plé, L. Lagardère, S. Huppert, and J.-P. Piquemal, "Improving condensed-phase water dynamics with explicit nuclear quantum effects: the polarizable q-amoeba force field", The Journal of Physical Chemistry B **126**, PMID: 36270033, 8813–8826 (2022) (cit. on pp. 22, 25).

[76]T. Plé, N. Mauger, O. Adjoua, et al., "Routine molecular dynamics simulations including nuclear quantum effects: from force fields to machine learning potentials", Journal of Chemical Theory and Computation **19**, PMID: 36856658, 1432–1445 (2023) (cit. on pp. 22, 87).

[77]M. E. Tuckerman, *Statistical mechanics : theory and molecular simulation*, eng, Oxford, 2010 (cit. on p. 24).

[78]D. Chandler, *Introduction to modern statistical mechanics* (Oxford University Press, 1987) (cit. on p. 24).

[79]P. G. Wolynes, "Imaginary time path integral Monte Carlo route to rate coefficients for nonadiabatic barrier crossing", The Journal of Chemical Physics **87**, 6559–6561 (1987) (cit. on p. 24).

[80]T. Plé, "Nuclear quantum dynamics : exploration and comparison of trajectory-based methods", 2020SORUS413, PhD thesis (2020) (cit. on p. 24).

[81]E. Mangaud, S. Huppert, T. Plé, et al., "The fluctuation–dissipation theorem as a diagnosis and cure for zero-point energy leakage in quantum thermal bath simulations", Journal of Chemical Theory and Computation **15**, PMID: 30939002, 2863–2880 (2019) (cit. on pp. 25, 26).

[82]T. Plé, L. Lagardère, and J.-P. Piquemal, *Force-field-enhanced neural network interactions: from local equivariant embedding to atom-in-molecule properties and long-range effects*, 2023 (cit. on pp. 25, 38).

[83]K. J. Bowers, E. Chow, H. Xu, et al., "Scalable algorithms for molecular dynamics simulations on commodity clusters", in Proceedings of the 2006 acm/ieee conference on supercomputing, SC '06 (2006), 84–es (cit. on p. 27).

[84]S. Le Grand, A. W. Götz, and R. C. Walker, "Spfp: speed without compromise—a mixed precision model for gpu accelerated molecular dynamics simulations", Computer Physics Communications **184**, 374–380 (2013) (cit. on p. 29).

[85]T. M. Mitchell, *Machine learning*, Vol. 1, 9 (McGraw-hill New York, 1997) (cit. on pp. 29, 30).

[86]I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, http://www.deeplearningbook.org (MIT Press, 2016) (cit. on pp. 29, 30, 33).

[87]P. Gkeka, G. Stoltz, A. Barati Farimani, et al., "Machine learning force fields and coarse-grained variables in molecular dynamics: application to materials and biological systems", Journal of Chemical Theory and Computation **16**, PMID: 32559068, 4757–4775 (2020) (cit. on p. 31).

[88]Z. Belkacemi, P. Gkeka, T. Lelièvre, and G. Stoltz, "Chasing collective variables using autoencoders and biased trajectories", Journal of Chemical Theory and Computation **18**, PMID: 34965117, 59–78 (2022) (cit. on p. 31).

[89]A. Mardt, L. Pasquali, H. Wu, and F. Noé, "Vampnets for deep learning of molecular kinetics", Nature Communications **9**, 5 (2018) (cit. on pp. 31, 197).

[90]D. P. Kingma and M. Welling, *Auto-encoding variational bayes*, 2022 (cit. on p. 31).

[91]R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates", in Advances in knowledge discovery and data mining, edited by J. Pei, V. S. Tseng, L. Cao, H. Motoda, and G. Xu (2013), pp. 160–172 (cit. on p. 32).

[92]M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: ordering points to identify the clustering structure", in Proceedings of the 1999 acm sigmod international conference on management of data, SIGMOD '99 (1999), pp. 49–60 (cit. on p. 32).

[93]K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?", in 2009 ieee 12th international conference on computer vision (2009), pp. 2146–2153 (cit. on p. 33).

[94]V. L. Deringer, A. P. Bartók, N. Bernstein, et al., "Gaussian process regression for materials and molecules", Chemical Reviews **121**, 10073–10141 (2021) (cit. on p. 34).

[95]S. Chmiela, A. Tkatchenko, H. E. Sauceda, et al., "Machine learning of accurate energy-conserving molecular force fields", Science Advances **3**, e1603015 (2017) (cit. on p. 34).

[96]K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, et al., *Schnet: a continuous-filter convolutional neural network for modeling quantum interactions*, 2017 (cit. on p. 34).

[97]T. B. Blank, S. D. Brown, A. W. Calhoun, and D. J. Doren, "Neural network models of potential energy surfaces", The Journal of Chemical Physics **103**, 4129–4137 (1995) (cit. on p. 34).

[98]J. Behler, "Atom-centered symmetry functions for constructing high-dimensional neural network potentials", The Journal of Chemical Physics **134**, 074106, 10.1063/1.3553717 (2011) (cit. on p. 35).

[99]J. S. Smith, O. Isayev, and A. E. Roitberg, "Ani-1: an extensible neural network potential with dft accuracy at force field computational cost", Chem. Sci. **8**, 3192–3203 (2017) (cit. on pp. 36, 103).

[100]J. S. Smith, O. Isayev, and A. E. Roitberg, "Ani-1, a data set of 20 million calculated off-equilibrium conformations for organic molecules", Scientific Data **4**, 170193 (2017) (cit. on pp. 36, 39).

[101]J. S. Smith, R. Zubatyuk, B. Nebgen, et al., "The ani-1ccx and ani-1x data sets, coupled-cluster and density functional theory properties for molecules", Scientific Data **7**, 134 (2020) (cit. on pp. 36, 103).

[102]C. Devereux, J. S. Smith, K. K. Huddleston, et al., "Extending the applicability of the ani deep learning molecular potential to sulfur and halogens", Journal of Chemical Theory and Computation **16**, 4192–4202 (2020) (cit. on pp. 36, 67, 103).

[103]L. Zhang, J. Han, H. Wang, R. Car, and W. E, "Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics", Phys. Rev. Lett. **120**, 143001 (2018) (cit. on pp. 37, 103).

[104]H. Wang, L. Zhang, J. Han, and W. E, "Deepmd-kit: a deep learning package for many-body potential energy representation and molecular dynamics", Computer Physics Communications **228**, 178–184 (2018) (cit. on pp. 37, 103).

[105]W. Jia, H. Wang, M. Chen, et al., "Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning", in Sc20: international conference for high performance computing, networking, storage and analysis (IEEE, 2020), pp. 1–14 (cit. on pp. 37, 67).

[106]K. T. Schütt, O. T. Unke, and M. Gastegger, *Equivariant message passing for the prediction of tensorial properties and molecular spectra*, 2021 (cit. on p. 38).

[107]O. T. Unke, S. Chmiela, M. Gastegger, et al., "Spookynet: learning force fields with electronic degrees of freedom and nonlocal effects", Nature Communications **12**, 7273 (2021) (cit. on p. 38).

[108]S. Batzner, A. Musaelian, L. Sun, et al., "E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials", Nature Communications **13**, 2453 (2022) (cit. on p. 38).

[109]A. Musaelian, S. Batzner, A. Johansson, et al., *Learning local equivariant representations for large-scale atomistic dynamics*, 2022 (cit. on p. 38).

[110]P. Eastman, P. K. Behara, D. L. Dotson, et al., "Spice, a dataset of drug-like molecules and peptides for training machine learning potentials", Scientific Data **10**, 11 (2023) (cit. on p. 40).

[111]A. G. Donchev, A. G. Taube, E. Decolvenaere, et al., "Quantum chemical benchmark databases of gold-standard dimer interaction energies", Scientific Data **8**, 55 (2021) (cit. on p. 41).

[112]J. He
nin, T. Lelie
vre, M. R. Shirts, O. Valsson, and L. Delemotte, "Enhanced sampling methods for molecular dynamics simulations [article v1.0]", Living Journal of Computational Molecular Science **4** (2022) (cit. on p. 42).

[113]C. H. Bennett, "Efficient estimation of free energy differences from monte carlo data", Journal of Computational Physics **22**, 245–268 (1976) (cit. on p. 44).

[114]X. Ding, J. Z. Vilseck, and C. L. Brooks, "Fast solver for large scale multistate bennett acceptance ratio equations", J. Chem. Theory. Comput. **15**, 799–802 (2019) (cit. on p. 44).

[115]M. R. Shirts and J. D. Chodera, "Statistically optimal analysis of samples from multiple equilibrium states", The Journal of Chemical Physics **129**, 124105, 10.1063/1.2978177 (2008) (cit. on p. 44).

[116]M. R. Shirts and J. D. Chodera, "Statistically optimal analysis of samples from multiple equilibrium states", The Journal of Chemical Physics **129**, 124105, 10.1063/1.2978177 (2008) (cit. on p. 44).

[117]C. Jarzynski, "Nonequilibrium equality for free energy differences", Phys. Rev. Lett. **78**, 2690–2693 (1997) (cit. on p. 45).

[118]T.-S. Lee, B. K. Allen, T. J. Giese, et al., "Alchemical binding free energy calculations in amber20: advances and best practices for drug discovery", Journal of Chemical Information and Modeling **60**, PMID: 32936637, 5595–5623 (2020) (cit. on p. 45).

[119]M. Harger, D. Li, Z. Wang, et al., "Tinker-openmm: absolute and relative alchemical free energies using amoeba on gpus", Journal of Computational Chemistry **38**, 2047–2055 (2017) (cit. on p. 45).

[120]G. Torrie and J. Valleau, "Nonphysical sampling distributions in monte carlo free-energy estimation: umbrella sampling", Journal of Computational Physics **23**, 187–199 (1977) (cit. on p. 45).

[121]D. Hamelberg, J. Mongan, and J. A. McCammon, "Accelerated molecular dynamics: A promising and efficient simulation method for biomolecules", The Journal of Chemical Physics **120**, 11919–11929 (2004) (cit. on p. 46).

[122]D. Hamelberg, C. A. F. de Oliveira, and J. A. McCammon, "Sampling of slow diffusive conformational transitions with accelerated molecular dynamics", The Journal of Chemical Physics **127**, 155102, 10.1063/1.2789432 (2007) (cit. on p. 46).

[123]Y. Miao, F. Feixas, C. Eun, and J. A. McCammon, "Accelerated molecular dynamics simulations of protein folding", J. Comput. Chem. **36**, 1536–1549 (2015) (cit. on p. 46).

[124]Y. Miao, V. A. Feher, and J. A. McCammon, "Gaussian accelerated molecular dynamics: unconstrained enhanced sampling and free energy calculation", J. Chem. Theory. Comput. **11**, 3584–3595 (2015) (cit. on p. 47).

[125]Y. Miao and J. A. McCammon, "Gaussian accelerated molecular dynamics: theory, implementation, and applications", in *Annu. rep. comput. chem.* Vol. 13 (Elsevier, 2017), pp. 231–278 (cit. on p. 47).

[126]J. Wang, P. R. Arantes, A. Bhattarai, et al., "Gaussian accelerated molecular dynamics: principles and applications", Wiley Interdiscip. Rev. Comput. Mol. Sci., e1521 (2021) (cit. on p. 47).

[127]Y.-T. Wang and Y.-H. Chan, "Understanding the molecular basis of agonist/antagonist mechanism of human mu opioid receptor through gaussian accelerated molecular dynamics method", Sci. Rep. **7**, 1–11 (2017) (cit. on p. 47).

[128]G. Palermo, "Structure and dynamics of the crispr–cas9 catalytic complex", J. Chem. Inf. Model. **59**, 2394–2406 (2019) (cit. on p. 47).

[129]Y. Miao, A. Bhattarai, and J. Wang, "Ligand gaussian accelerated molecular dynamics (ligamd): characterization of ligand binding thermodynamics and kinetics", J. Chem. Theory. Comput. **16**, 5526–5547 (2020) (cit. on p. 47).

[130]H. Oshima, S. Re, and Y. Sugita, "Replica-exchange umbrella sampling combined with gaussian accelerated molecular dynamics for free-energy calculation of biomolecules", J. Chem. Theory. Comput. **15**, 5199–5208 (2019) (cit. on p. 47).

[131]J. Wang and Y. Miao, "Peptide gaussian accelerated molecular dynamics (pep-gamd): enhanced sampling and free energy and kinetics calculations of peptide binding", J. Chem. Phys. **153**, 154109 (2020) (cit. on p. 47).

[132]A. Laio and M. Parrinello, "Escaping free-energy minima", Proceedings of the National Academy of Sciences **99**, 12562–12566 (2002) (cit. on p. 47).

[133]A. Barducci, G. Bussi, and M. Parrinello, "Well-tempered metadynamics: a smoothly converging and tunable free-energy method", Phys. Rev. Lett. **100**, 020603 (2008) (cit. on p. 47).

[134]O. Valsson, P. Tiwary, and M. Parrinello, "Enhancing important fluctuations: rare events and metadynamics from a conceptual viewpoint", Annual Review of Physical Chemistry **67**, PMID: 26980304, 159–184 (2016) (cit. on p. 47).

[135]N. S. Hinrichs and V. S. Pande, "Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics", The Journal of Chemical Physics **126**, 244101, 10.1063/1.2740261 (2007) (cit. on p. 49).

[136]X. Huang, G. R. Bowman, S. Bacallado, and V. S. Pande, "Rapid equilibrium sampling initiated from nonequilibrium data", Proceedings of the National Academy of Sciences **106**, 19765–19769 (2009) (cit. on p. 49).

[137]V. A. Voelz, B. Elman, A. M. Razavi, and G. Zhou, "Surprisal metrics for quantifying perturbed conformational dynamics in markov state models", Journal of Chemical Theory and Computation **10**, PMID: 26583253, 5716–5728 (2014) (cit. on p. 49).

[138]E. Chiavazzo, R. Covino, R. R. Coifman, et al., "Intrinsic map dynamics exploration for uncharted effective free-energy landscapes", Proceedings of the National Academy of Sciences **114**, E5494–E5503 (2017) (cit. on p. 49).

[139]O. Kukharenko, K. Sawade, J. Steuer, and C. Peter, "Using dimensionality reduction to systematically expand conformational sampling of intrinsically disordered peptides", Journal of Chemical Theory and Computation **12**, PMID: 27588692, 4726–4734 (2016) (cit. on p. 50).

[140]Y. Sugita, A. Kitao, and Y. Okamoto, "Multidimensional replica-exchange method for free-energy calculations", The Journal of Chemical Physics **113**, 6042–6051 (2000) (cit. on p. 50).

[141]H. Oshima, S. Re, and Y. Sugita, "Replica-exchange umbrella sampling combined with gaussian accelerated molecular dynamics for free-energy calculation of biomolecules", Journal of Chemical Theory and Computation **15**, PMID: 31539245, 5199–5208 (2019) (cit. on p. 50).

[142]P. Gkeka, G. Stoltz, A. Barati Farimani, et al., "Machine learning force fields and coarse-grained variables in molecular dynamics: application to materials and biological systems", Journal of Chemical Theory and Computation **16**, 4757–4775 (2020) (cit. on p. 50).

[143]B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis", in Artificial neural networks — icann'97, edited by W. Gerstner, A. Germond, M. Hasler, and J.-D. Nicoud (1997), pp. 583–588 (cit. on p. 51).

[144]W. Chen and A. L. Ferguson, "Molecular enhanced sampling with autoencoders: on-the-fly collective variable discovery and accelerated free energy landscape exploration", Journal of Computational Chemistry **39**, 2079–2102 (2018) (cit. on p. 51).

[145]Z. Belkacemi, P. Gkeka, T. Lelièvre, and G. Stoltz, "Chasing collective variables using autoencoders and biased trajectories", Journal of Chemical Theory and Computation **18**, 59–78 (2022) (cit. on p. 51).

[146]J. M. L. Ribeiro, P. Bravo, Y. Wang, and P. Tiwary, "Reweighted autoencoded variational Bayes for enhanced sampling (RAVE)", The Journal of Chemical Physics **149**, 072301, 10.1063/1.5025487 (2018) (cit. on p. 51).

[147]Z. Shamsi, K. J. Cheng, and D. Shukla, "Reinforcement learning based adaptive sampling: reaping rewards by exploring protein conformational landscapes", The Journal of Physical Chemistry B **122**, PMID: 30126271, 8386–8395 (2018) (cit. on p. 51).

[148]C. Wehmeyer and F. Noé, "Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics", The Journal of Chemical Physics **148**, 241703, 10.1063/1.5011399 (2018) (cit. on p. 52).

[149]G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé, "Identification of slow molecular order parameters for Markov model construction", The Journal of Chemical Physics **139**, 015102, 10.1063/1.4811489 (2013) (cit. on pp. 52, 197).

[150]M. K. Scherer, B. Trendelkamp-Schroer, F. Paul, et al., "Pyemma 2: a software package for estimation, validation, and analysis of markov models", Journal of Chemical Theory and Computation **11**, PMID: 26574340, 5525–5542 (2015) (cit. on pp. 52, 197).

[151]A. Mardt, L. Pasquali, H. Wu, and F. Noé, "Vampnets for deep learning of molecular kinetics", Nature Communications **9**, 5 (2018) (cit. on p. 52).

[152]C. R. Schwantes and V. S. Pande, "Modeling molecular kinetics with tica and the kernel trick", Journal of Chemical Theory and Computation **11**, PMID: 26528090, 600–608 (2015) (cit. on p. 52).

[153]H. N. Do and Y. Miao, "Deep boosted molecular dynamics: accelerating molecular simulations with gaussian boost potentials generated using probabilistic bayesian deep neural network", The Journal of Physical Chemistry Letters **14**, PMID: 37219922, 4970–4982 (2023) (cit. on p. 52).

[154]A. Tkatchenko, R. A. DiStasio, R. Car, and M. Scheffler, "Accurate and efficient method for many-body van der waals interactions", Phys. Rev. Lett. **108**, 236402 (2012) (cit. on p. 55).

[155]P. P. Poier, L. Lagardère, and J.-P. Piquemal, "O(n) stochastic evaluation of many-body van der waals energies in large complex systems", Journal of Chemical Theory and Computation **18**, 1633–1645 (2022) (cit. on p. 55).

[156]T. C. Lillestolen and R. J. Wheatley, "Atomic charge densities generated using an iterative stockholder procedure", The Journal of Chemical Physics **131**, 144101 (2009) (cit. on p. 55).

[157]A. J. Misquitta, A. J. Stone, and F. Fazeli, "Distributed multipoles from a robust basis-space implementation of the iterated stockholder atoms procedure", Journal of Chemical Theory and Computation **10**, PMID: 26583224, 5405–5418 (2014) (cit. on p. 55).

[158]T. Verstraelen, S. Vandenbrande, F. Heidar-Zadeh, et al., "Minimal basis iterative stockholder: atoms in molecules for force-field development", Journal of Chemical Theory and Computation **12**, 3894–3912 (2016) (cit. on p. 55).

[159]J. Řezáč, K. E. Riley, and P. Hobza, "S66: a well-balanced database of benchmark interaction energies relevant to biomolecular structures", Journal of Chemical Theory and Computation **7**, 2427–2438 (2011) (cit. on p. 56).

[160]N. Gresh, G. A. Cisneros, T. A. Darden, and J.-P. Piquemal, "Anisotropic, polarizable molecular mechanics studies of inter-, intra-molecular interactions, and ligand-macromolecule complexes. a bottom-up strategy.", Journal of Chemical Theory and Computation **3**, 1960–1986 (2007) (cit. on p. 65).

[161]S. Naseem-Khan, L. Lagardère, C. Narth, et al., "Development of the quantum inspired sibfa many-body polarizable force field: enabling condensed phase molecular dynamics simulations", J. Chem. Theory. Comput. DOI: 10.1021/acs.jctc.2c00029 (2022) (cit. on p. 65).

[162]T. Jaffrelot Inizan, T. Plé, O. Adjoua, et al., "Scalable hybrid deep neural networks/polarizable potentials biomolecular simulations including long-range effects", Chem. Sci. **14**, 5438–5452 (2023) (cit. on p. 67).

[163]V. Kapil, M. Rossi, O. Marsalek, et al., "i-PI 2.0: a universal force engine for advanced molecular simulations", Comput. Phys. Commun. **236**, 214–223 (2019) (cit. on p. 87).

[164]A. P. Thompson, H. M. Aktulga, R. Berger, et al., "LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales", Comput. Phys. Commun. **271**, 108171 (2022) (cit. on p. 87).

[165]W. Fang, J. Chen, M. Rossi, et al., "Inverse temperature dependence of nuclear quantum effects in dna base pairs", J. Phys. Chem. Lett. **7**, 2125–2131 (2016) (cit. on p. 103).

[166]Y. Law and A. Hassanali, "The importance of nuclear quantum effects in spectral line broadening of optical spectra and electrostatic properties in aromatic chromophores", J. Chem. Phys. **148**, 102331 (2018) (cit. on p. 103).

[167]X.-Z. Li, B. Walker, and A. Michaelides, "Quantum nature of the hydrogen bond", Proc. Natl. Acad. Sci. USA **108**, 6369–6373 (2011) (cit. on p. 103).

[168]T. P. Senftle, S. Hong, M. M. Islam, et al., "The reaxff reactive force-field: development, applications and future directions", npj Computational Materials **2**, 15011 (2016) (cit. on p. 105).

[169]O. T. Unke and M. Meuwly, "Physnet: a neural network for predicting energies, forces, dipole moments, and partial charges", Journal of Chemical Theory and Computation **15**, PMID: 31042390, 3678–3693 (2019) (cit. on p. 105).

[170]Z.-H. Xu and M. Meuwly, "Multistate reactive molecular dynamics simulations of proton diffusion in water clusters and in the bulk", The Journal of Physical Chemistry B **123**, PMID: 31647873, 9846–9861 (2019) (cit. on p. 105).

[171]T. Jaffrelot Inizan, F. Célerse, O. Adjoua, et al., "High-resolution mining of the sars-cov-2 main protease conformational space: supercomputer-driven unsupervised adaptive sampling", Chem. Sci. **12**, 4889–4907 (2021) (cit. on p. 109).

[172]D. El Ahdab, L. Lagardère, T. J. Inizan, et al., "Interfacial water many-body effects drive structural dynamics and allosteric interactions in sars-cov-2 main protease dimerization interface", The Journal of Physical Chemistry Letters **12**, PMID: 34196568, 6218–6226 (2021) (cit. on p. 131).

[173]F. Célerse, T. J. Inizan, L. Lagardère, et al., "An efficient gaussian-accelerated molecular dynamics (gamd) multilevel enhanced sampling strategy: application to polarizable force fields simulations of large biological systems", Journal of Chemical Theory and Computation **18**, PMID: 35080892, 968–977 (2022) (cit. on p. 144).

[174]L. El Khoury, Z. Jing, A. Cuzzolin, et al., "Computationally driven discovery of sars-cov-2 mpro inhibitors: from design to experimental validation", Chem. Sci. **13**, 3674–3687 (2022) (cit. on p. 197).

A

Enhancing Molecular Dynamics Simulations: Leveraging Tinker-HP and GPU Acceleration for Improved Performance

Introduction

To meet the increasing computational demands for longer simulations of larger biomolecular systems, it has been essential to employ parallelization and acceleration strategies. Tinker-HP was initially developed as a massively MPI parallel package dedicated to accelerating various FFs, especially PFFs, and has proven to be highly efficient, scaling up to tens of thousands of CPUs on modern petascale supercomputers. However, recent years have seen the emergence of GPUs which offer impressive computational power compared to CPUs. The present article aims to achieve two goals: designing an efficient, native Tinker-HP GPU implementation with lower and double precision arithmetic, and optimizing it for HPC in the massively parallel context of modern multi-GPU pre-exascale supercomputer systems.[13]

JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

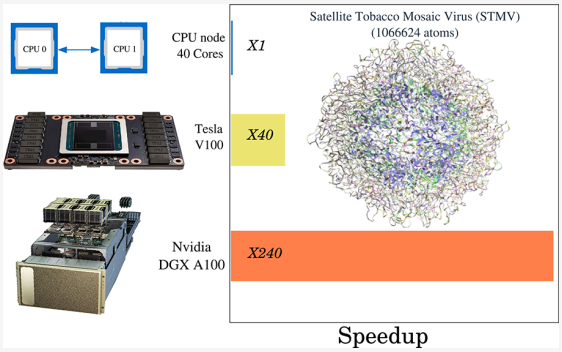
Tinker-HP: Accelerating Molecular Dynamics Simulations of Large Complex Systems with Advanced Point Dipole Polarizable Force Fields Using GPUs and Multi-GPU Systems

Olivier Adjoua, Louis Lagardère,* Luc-Henri Jolly, Arnaud Durocher, Thibaut Very, Isabelle Dupays, Zhi Wang, Théo Jaffrelot Inizan, Frédéric Célerse, Pengyu Ren, Jay W. Ponder, and Jean-Philip Piquemal*

Cite This: *J. Chem. Theory Comput.* 2021, 17, 2034−2053 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: We present the extension of the Tinker-HP package (Lagardère, et al. *Chem. Sci.* **2018**, *9*, 956−972) to the use of Graphics Processing Unit (GPU) cards to accelerate molecular dynamics simulations using polarizable many-body force fields. The new high-performance module allows for an efficient use of single- and multiple-GPU architectures ranging from research laboratories to modern supercomputer centers. After detailing an analysis of our general scalable strategy that relies on OpenACC and CUDA, we discuss the various capabilities of the package. Among them, the multiprecision possibilities of the code are discussed. If an efficient double precision implementation is provided to preserve the possibility of fast reference computations, we show that a lower precision arithmetic is preferred providing a similar accuracy for molecular dynamics while exhibiting superior performances. As Tinker-HP is mainly dedicated to accelerate simulations using new generation point dipole polarizable force field, we focus our study on the implementation of the AMOEBA model. Testing various NVIDIA platforms including 2080Ti, 3090, V100, and A100 cards, we provide illustrative benchmarks of the code for single- and multicards simulations on large biosystems encompassing up to millions of atoms. The new code strongly reduces time to solution and offers the best performances to date obtained using the AMOEBA polarizable force field. Perspectives toward the strong-scaling performance of our multinode massive parallelization strategy, unsupervised adaptive sampling and large scale applicability of the Tinker-HP code in biophysics are discussed. The present software has been released in phase advance on GitHub in link with the High Performance Computing community COVID-19 research efforts and is free for Academics (see https://github.com/TinkerTools/tinker-hp).



INTRODUCTION

Molecular dynamics (MD) is a very active research field that is continuously progressing.[1,2] Among various evolutions, the definition of force fields themselves grows more complex. Indeed, beyond the popular pairwise additive models[3−7] that remain extensively used, polarizable force field (PFF) approaches are becoming increasingly mainstream and start to be more widely adopted,[8−11] mainly because accounting for polarizability is often crucial for complex applications and adding new physics to the model through the use of many-body potentials can lead to significant accuracy enhancements.[10] Numerous approaches are currently under development but a few methodologies such as the Drude[12−14] or the AMOEBA[15−17] models emerge. These models are more and more employed because of the alleviation of their main bottleneck: their larger computational cost compared to classical pairwise models. Indeed, the availability of High Performance Computing (HPC) implementations of such models within popular packages such as NAMD[18] or GROMACS[19] for Drude or Tinker-HP[20] for AMOEBA fosters the diffusion of these new generation techniques within the research community. This paper is dedicated to the evolution of the Tinker-HP package.[20] The software, which is part of the Tinker distribution,[21] was initially introduced as a double precision massively parallel message passing interface (MPI) addition to Tinker dedicated to the acceleration of the various PFFs and nonpolarizable force fields (n-PFFs) present within

Received: November 6, 2020
Published: March 23, 2021

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

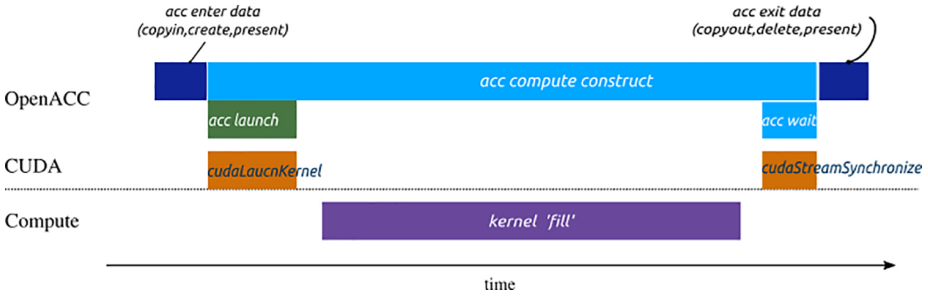


Figure 1. OpenACC synchronous execution model on test kernel <fill>.

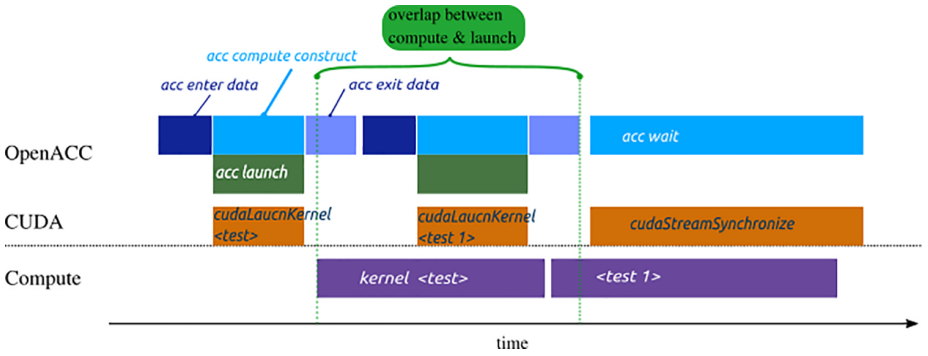


Figure 2. OpenACC asynchronous execution on both kernels <test> and <test 1>.

the Tinker package. The code was shown to be really efficient, being able to scale on up to tens of thousand cores on modern petascale supercomputers.[20,22] Recently, it has been optimized on various platforms taking advantage of vectorization and of the evolution of the recent CPUs (Central Processing Units).[22] However, in the last 15 years, the field has been increasingly using GPUs (Graphic Processor Units)[23−25] taking advantage of low precision arithmetic. Indeed, such platforms offer important computing capabilities at both low cost and high energy efficiency allowing for reaching routine microsecond simulations on standard GPU cards with pair potentials.[24,26] Regarding the AMOEBA polarizable force field, the OpenMM package[27] was the first to propose an AMOEBA-GPU library that was extensively used within Tinker through the Tinker-OpenMM GPU interface.[28] The present contribution aims to address two goals: (i) the design of an efficient native Tinker-HP GPU implementation; (ii) the HPC optimization in a massively parallel context to address both the use of research laboratories clusters and modern multi-GPU pre-exascale supercomputer systems. The paper is organized as follows. First, we will describe our OpenACC port and its efficiency in double precision. After observing the limitations of this implementation regarding the use of single precision, we introduce a new CUDA approach and detail the various parts of the code it concerns after a careful study of the precision. In both cases, we present benchmarks of the new code on illustrative large biosystems of increasing size on various NVIDIA platforms (including RTX 2080Ti, 3090, Tesla V100 and A100 cards). Then, we explore how to run on even larger systems and optimize memory management by making use of latest tools such as NVSHMEM.[29]

OPENACC APPROACH

Global Overview and Definitions. Tinker-HP is a molecular dynamics application with a MPI layer allowing a significant acceleration on CPUs. The core of the application is based on the resolution of the classical newton equations[20,30] given an interaction potential (force field) between atoms. In practice, a molecular dynamic simulation consists of the repetition of the call to an integrator routine defining the changes of the positions and the velocities of all the atoms of the simulated system between two consecutive time steps. The same process is repeated as many times as needed until the simulation duration is reached (see Figure 3). To distribute computations over the processes, a traditional three-dimensional domain decomposition is performed on the simulation box ($\boldsymbol{\Omega}$), which means that it is divided in subdomains ($\boldsymbol{\psi}$), each of which is associated with a MPI process. Then, within each time step, positions of the atoms and forces are exchanged between processes before and after the computation of the forces. Additionally, small communications are required after the update of the positions to deal with the fact that an atom can change the subdomain during a time step. This workflow is described in detail in ref 20.

In recent years a new paradigm has emerged to facilitate computation and programming on GPU devices. In the rest of the text, we will denote as *kernel* the smallest piece of code made of instructions designed for a unique purpose. Thus, a succession of kernels might constitute a *routine* and a *program* can be seen as a collection of routines designed for a specific purpose. There are two types of kernels

- *Serial* kernels, mostly used for variable configuration
- *Loops* kernels, operating on multiple data sets

This programming style, named OpenACC,[31,32] is a directive-based language similar to the multithreading OpenMP paradigm with an additional complexity level. Since a target kernel is destined to be executed on GPUs, it becomes crucial to manage data between both GPU and CPU platforms. At the most elementary level, OpenACC compiler interacts on

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

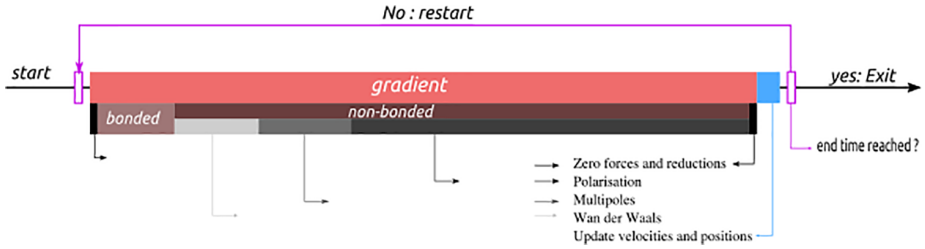


Figure 3. illustration of a MD time step.

a standard host (CPU) kernel and generates a device (GPU) kernel using directives implemented to describe its parallelism along with clauses to manage global data behavior at both entry and exit point and/or kernel launch configuration (Figure 1). This method offers two major benefits. Unlike the low-level CUDA programming language,[33] it takes only a few directives to generate a device kernel. Second, the same kernel is compatible with both platforms, CPUs and GPUs. The portability along with all the associated benefits such as host debug is therefore ensured. However, there are some immediate drawbacks mainly because CPUs and GPUs do not share the same architecture, specifications, and features. Individual CPU cores benefit from a significant optimization for serial tasks, a high clock frequency and integrated vectorization instructions to increase processing speed. GPUs on the other hand were developed and optimized from the beginning for parallel tasks with numerous aggregations of low clock cores holding multiple threads. This means that it may be necessary to reshape kernels to fit device architecture in order to get appropriate acceleration. Once we clearly exhibit a kernel parallelism and associate OpenACC directives to offload it on a device, it should perform almost as well as if it had been directly written in native CUDA. Still, in addition to kernel launch instruction (performed by both OpenACC and CUDA) before the appropriate execution, there is a global data checking operation overhead that might slow down execution (Figure 1). However, it is possible to overlap this operation using asynchronous device streams in the kernel configuration (Figure 2). Under proper conditions and with directly parallel kernels, OpenACC can already lead to an efficient acceleration close to the one reachable with CUDA.

In the following, we will say that a kernel is *semiparallel* if one can find a partition inside the instructions sequence that does not share any dependency at all. A semiparallel kernel is consequently defined *parallel* if all instructions in the partition do not induce a race condition within its throughput.

Once a kernel is device compiled, its execution requires a configuration defining the associated resources provided by the device. With OpenACC, these resources are respectively the total number of threads and the assignment stream. We can access the first one through the *gang* and *vector* clauses attached to a device compute region directive. A `gang` is a collection of vectors inside of which every thread can share cache memory. All gangs run separately on device streaming multiprocessors (SM) to process kernel instructions inside a stream where many other kernels are sequentially queued. OpenACC offers an intermediate parallelism level between gang and vector called `worker`. This level can be seen as a gang subdivision.

It is commonly known that GPUs are inefficient for sequential execution due to their latency. To cover up latency, each SM comes with a huge register file and cache memory in order to hold and run as many vectors as possible at the same time. Instructions from different gangs are therefore pipe-lined and injected in the compute unit.[33,34] From this emerges the kernel occupancy's concept, which is defined as the ratio between the gang's number concurrently running on one SM and the maximum gang number that can actually be held by this SM.

Global Scheme. The *parallel* computing power of GPUs is in constant evolution and the number of streaming multiprocessors (SM) is almost doubling with every generation. Considering their impressive compute potential in comparison to CPUs, one can assume that the only way to entirely benefit from this power is to offload the entire application on device. Any substantial part of the workflow of Tinker-HP should not be performed on the CPU platform. It will otherwise represent a bottleneck to performance in addition to requiring several data transfers. As for all MD applications, most of the computation lies in the evaluation of the forces. For the AMOEBA polarizable model, it takes around 97% of a time step to evaluate those forces when running sequentially on CPU platform. Of these 97%, around 10% concern bonded forces and 90% the nonbonded ones, namely, polarization, (multipolar) permanent electrostatics, and van der Waals. The polarization, which includes the iterative resolution of induced dipoles, largely dominates this part (see Figure 3). Nonbonded forces and polarization in particular will thus be our main focus regarding the porting and optimization. We will then benefit from the already present Tinker-HP MPI layer[20,22] to operate on several GPUs. The communications can then be made directly between GPUs by using a CUDA aware MPI implementation.[35] The Smooth Particle Mesh Ewald method[20,36,37] is at the heart of both the permanent electrostatics and polarization nonbonded forces used in Tinker-HP, first through the iterative solution of the induced dipoles and then through the final force evaluation. It consists in separating the electrostatic energy in two independent pieces: real space and reciprocal space contributions. Let us describe our OpenACC strategy regarding those two terms.

Real Space Scheme. Because the real space part of the total PME energy and forces has the same structure as the van der Waals one, the associated OpenACC strategy is the same. Evaluating real space energy and forces is made through the computation of pairwise interactions. Considering n atoms, a total of $n(n-1)$ pairwise interactions need to be computed. This number is reduced by half because of the symmetry of the interactions. Besides, because we use a cutoff distance after which we neglect these interactions, we can reduce their number to being proportional to n in homogeneous systems by using neighbor lists. The up-bound constant is naturally

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

Chart 1. OpenACC Real Space Offload Scheme[a]

```
c$acc parallel loop gang default(present) async
c$acc&         private(scaling_data)
do i = 1,numLocalAtoms
   iglob = glob(i)  ! Get Atom i global id
   !Get Atom iglob parameter and positions
   ...
   !Gather Atoms iglob scaling interactions in 'scaling_data'
   ...
c$acc loop vector
   do k = 1, numNeig(i)
      kglob = glob( list(k,i) )
      ! Get Atom kglob parameter and positions
      ! Compute distance (d) between iglob & kglob
      if (d < dcut) then
         call resolve_scaling_factor(scaling_data)
         ...
         call Compute_interaction !inlined
         ...
         call Update_(energy,forces,virial)
      end if
   end do
end do
```

[a]The kernel is offloaded onto a device using two of the three parallelism levels offered by OpenACC. The first loop is broken down over gangs and gathers all data related to atom `iglob` using gang's shared memory through the private clause. OpenACC vectors are responsible of the evaluation and the addition of forces and energy after resolving scaling factor if necessary. Regarding data management we make sure with the present clause that everything is available on device before the execution of the kernel.

reduced to a maximum neighbors for every atoms noted as $Neig_{max}$.

The number of interactions is up-bounded by n *$Neig_{max}$. In terms of implementation, we have written the compute algorithm into a single loop kernel. As all the interactions are independent, the kernel is semiparallel regarding each iteration. By making sure that energy and forces are added one at a time, the kernel becomes parallel. To do that, we can use atomic operations on GPUs, which allow us to make this operation in parallel and solve any race condition issue without substantially impacting parallel performance. By doing so, real space kernels looks like Chart 1.

At first, our approach was designed to directly offload the CPU vectorized real space compute kernels that use small arrays to compute pairwise interactions in hopes of aligning the memory access pattern at the vector level and therefore accelerate the code.[22] This requires each gang to privatize every temporary array and results in a significant overhead with memory reservation associated with a superior bound on the gang's number. Making interactions computation scalar helps us remove those constraints and double the kernel performance. The explanation behind this increase arises from the use of GPU scalar registers. Still, one has to resolve the scaling factors of every interactions. As it happens inside gang shared memory, the performance is slightly affected. However, we would benefit from a complete removal of this inside search. There are two potential drawbacks to this approach:

- Scaling interactions between neighboring atoms of the same molecule can become very complex. This is particularly true with large proteins. Storage workspace can potentially affect shared memory and also kernel's occupancy.
- Depending on the interactions, there is more than one kind of scaling factor. For example, every AMOEBA polarization interaction needs three different scaling factors.

The best approach is then to compute scaling interactions separately in a second kernel. Because they only involve connected atoms, their number is small compared to the total number of nonbonded interactions. We first compute unscaled nonbonded interactions and then apply scaling correction in a second part. An additional issue is to make this approach compatible with the 3d domain decomposition. Our previous kernel then reads as in Chart 2.

Reciprocal Space Scheme. The calculation of Reciprocal space PME interactions essentially consists in five steps:

1. interpolating the (multipolar) density of charge at stake on a 3D grid with flexible b-spline order (still, the

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

Chart 2. Final OpenACC Real Space Offload Scheme[a]

```
c$acc parallel loop gang default(present) async
do i = 1,numLocalAtoms
   iglob = glob(i)  ! Get Atom i global id
   !Get Atom iglob parameter and positions
   ...
c$acc loop vector
   do k = 1, numNeig(i)
      kglob = glob( list(k,i) )
      ! Get Atom kglob parameter and positions
      ! Compute distance (d) between iglob & kglob
      if (d < dcut) then
         call Compute_interaction !inlined
         ...
         call Update_(energy,forces,virial)
      end if
   end do
end do

call correct_scaling
```

[a]This kernel is more balanced and exposes a much more computational load over vectors. A "correct_scaling" routine applies the correction of the scaling factors. This procedure appears to be much more suitable to device execution.

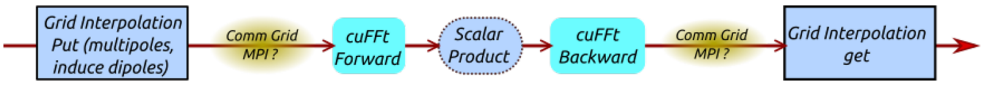


Figure 4. Reciprocal space offload scheme. Charge interpolation and Force interpolation are both written in a single kernel. They are naturally parallel except for the atomic contributions to the grid in the first one. The approach remains the same for data management between host and device as for real space: all data are by default device resident to prevent any sort of useless transfer. Regarding MPI communications, exchanges take place directly between GPUs through interconnection.

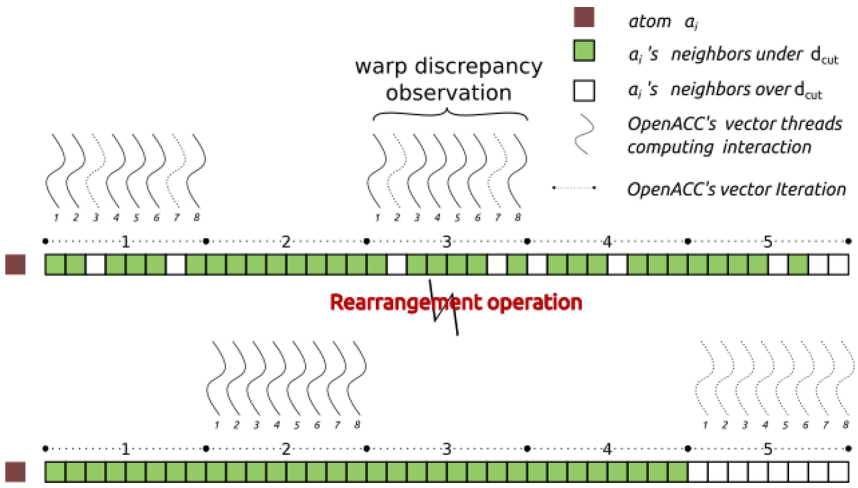


Figure 5. Illustration of compute balance due to the list reordering. Unbalanced computation in the first image induces an issue called warp discrepancy: a situation where all threads belonging to the same vector do not follow the same instructions. Minimizing that can increase kernel performance significantly since we ensure load balancing among each thread inside the vector.

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

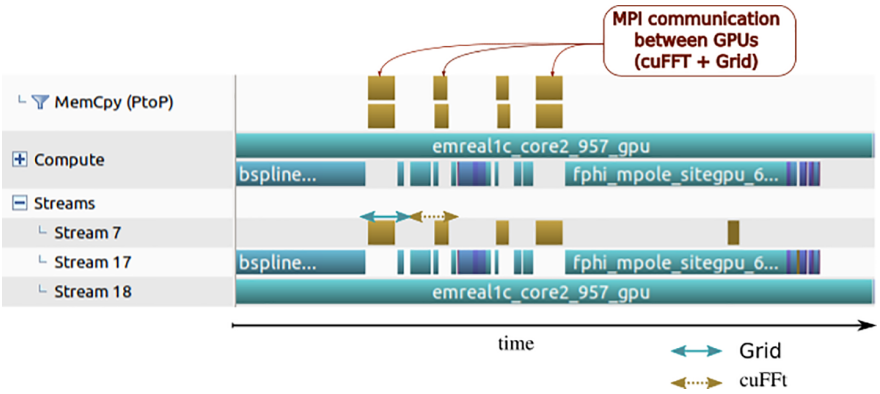


Figure 6. Representation of `cuFFt`'s communication/computation overlap using different streams for direct and reciprocal space. Real space computation kernels are assigned to asynchronous stream 18. Reciprocal ones go into high priority asynchronous stream 17. The real space kernel therefore recovers `FFt` grid exchanges. This profile was retrieved on 2 GPUs.

implementation is optimized to use an order of 5 as it is the default and the one typically used with AMOEBA).
2. switching to Fourier space by using a forward fast Fourier transform (`FFt`)
3. performing a trivial scalar product in reciprocal space
4. performing a backward `FFt` to switch back to real space
5. performing a final multiplication by b-splines to interpolate the reciprocal forces

Regarding parallelism, Tinker-HP uses a two-dimensional decomposition of the associated 3d grid based on successive 1D `FFt`s. Here, we use the `cuFFt` library.[38] The OpenACC offload scheme for reciprocal space is described in Figure 4.

We just reviewed our offload strategy of the nonbonded forces kernels with OpenACC, but the bonded ones remain to be treated. Also, the MPI layer has to be dealt with. The way bonded forces are computed is very similar to the real space ones, albeit simpler, which makes their offloading relatively straightforward. MPI layer kernels require, on the other hand, a slight rewriting as communications are made after a packing pretreatment. In parallel, one does not control the throughput order of this packing operation. This is why it becomes necessary to also communicate the atom list of each process to their neighbors. Now that we presented the main offload strategies, we can focus on some global optimizations regarding the implementation and execution for single and multiple GPUs. Some of them lead to very different results depending on the device architecture.

Optimizations Opportunities.

>A first optimization is to impose an optimal bound on the vector size when computing pair interactions. In a typical setup, for symmetry reasons, the number of neighbors for real space interactions varies between zero and a few hundred. Because of that second loop in Chart 2, the smallest vector length (32) is appropriate to balance computation among the threads it contains. Another optimization concerns the construction of the neighbor lists. Let us recall that it consists of storing, for every atom, the neighbors that are closer than a cut distance (d_{cut}) plus a buffer d_{buff}. This buffer is related to the frequency at which the list has to be updated. To balance computation at the vector level and at the same time reduce warp discrepancy (as illustrated in Figure 5), we have implemented a reordering kernel: we reorder the neighbor list for each atom so that the firsts are the ones under d_{cut} distance.

>A second optimization concerns the iterative resolution of the induced dipoles. Among the algorithms presented in refs 37 and 39, the first method we offloaded is the preconditioned conjugated gradient (PCG). It involves a (polarization) matrix-vector product at each iteration. Here, the idea is to reduce the computation and favor coalesce memory access by precomputing and storing the elements of (the real space part) of the matrix before the iterations. As the matrix-vector product is being repeated, we see a performance gain starting from the second iteration. This improves performance but implies a memory cost that could be an issue on large systems or on GPUs with small memory capabilities. This overhead will be reduced at a high level of multidevice parallelism.

>An additional improvement concerns the two-dimensional domain decomposition of the reciprocal space 3D grid involved with `FFt`. The parallel scheme for `FFt` used in Tinker-HP is the following for a forward transform:

$$\mathrm{FFT}(s)\ 1\mathrm{d}\ \dim(x) + x\ \text{Transpose}\ y + \mathrm{FFt}(s)\ 1\mathrm{d}\ \dim(y) + y\ \text{Transpose}\ z + \mathrm{FFT}(s)\ 1\mathrm{d}\ \dim(z)$$

Each transposition represents an all-to-all MPI communication, which is the major bottleneck preventing most MD applications using PME to scale across nodes.[20,22,40] Given the GPUs' huge computing power, this communication is even more problematic in that context. On the device, we use the `cuFFt`[38] library. Using many `cuFFt` 1d batches is not as efficient as using fewer batches in a higher dimension. Indeed, devices are known to underperform with low saturation kernels. In order to reduce MPI exchanges and increase load on device, we adopted a simple 3d dimensional cuFFt batch when running on a single device. On multiple GPUs, we use the following scheme based on a 1d domain decomposition along the z axis:

$$\mathrm{cuFFt}(s)\ 2\mathrm{d}\ \dim(x, y) + y\ \text{Transpose}\ z + \mathrm{cuFFt}(s)\ 1\mathrm{d}\ \dim(z)$$

which gives a 25% improvement compared to the initial approach.

>Profiling the application on a single device, we observed that real space computation is on average 5 times slower than reciprocal space computation. This trend reverses using multiple GPUs because of the communications mentioned above. This motivated the assignment of these two parts in two different priority streams. Reciprocal space kernels along with MPI communications are queued inside the higher priority

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

stream, and real space kernels, devoid of communications, can operate at the same time on the lower priority stream to recover communications. This is illustrated in Figure 6.

Simulation Validation and Benchmarks. Here, we use the same bench systems as in refs 20 and 22: the solvated DHFR protein, the solvated COX protein, and the STMV virus, all with the AMOEBA force field, respectively made of 23558, 171219, and 1066600 atoms. The molecular dynamics simulations were run in the *NVT* ensemble at 300 K for 5 ps simulation using a (bonded/nonbonded) RESPA integrator with a 2 fs outer time step (and a 1 fs inner time step)[41] and the Bussi thermostat.[42] The performance was averaged over the complete runs. For validation purposes, we compared the potential energy, temperature, and pressure of the systems during the first 50 time steps with values obtained with Tinker-HP v1.2. Furthermore, these results were compared to Tinker-OpenMM in the same exact setup.[28]

We can directly observe the technical superiority of the Quadro architecture compared to the Geforce one. Double precision (DP) compute units of the V100 allow us to vastly outperform the Geforce. In addition, by comparing the performance of the Geforce RTX to the one of the Quadro GV100, we see that Quadro devices are much less sensitive to warp discrepancy and noncoalesced data accessing pattern. It is almost as if the architecture of the V100 card overcomes traditional optimizations techniques related to parallel device implementation. However, we see that our pure OpenACC implementation manages to deliver more performance than usual device MD application with PFF in DP. The V100 results were obtained on the Jean-Zay HPE SGI 8600 cluster of the IDRIS supercomputer Center (GENCI-CNRS, Orsay, France) whose converged partitions are respectively made of 261 and 351 nodes. Each one is made of 2 Intel Cascade Lake 6248 processors (20 cores at 2.5 GHz) accelerated with 4 NVIDIA Tesla V100 SXM2 GPUs, interconnected through NVIDIA NVLink, and coming respectively with 32 GB of memory on the first partition and 16 GB on the second. Here as in all the tests presented in this paper, all the MPI communications were made with a CUDA aware MPI implementation.[35] This result is very satisfactory as a single V100 card is at least 10 times faster than an entire node of this supercomputer using only CPUs.

Multidevice benchmark results compared with host-platform execution are presented in Figure 7. In practice, the DHFR protein is too small to scale out. MPI communications overcome the computations even with optimizations. On the other hand, COX and STMV systems show good multi-GPU performances. Adding our latest MPI optimizations (FFt reshaping and asynchronous computing between the direct and reciprocal part of PME) allows for a substantial gain in performances. We see that on a Jean-Zay node we can only benefit from the maximum communication bandwidth when running on the entire node; hence the relative inflection point on the STMV performances on the 2 GPU setup. Indeed, all devices are interconnected inside one node in such a way that they all share the interconnection bandwidth. More precisely, running on 2 GPUs reduces the bandwidth by three and therefore affects the scalability. It is almost certain that results would get better on an interconnected node made exclusively of 2 GPUs. Those results are more than encouraging considering the fact that we manage to achieve them with a full OpenACC implementation of Tinker-HP (direct portage of the reference CPU code) in addition to some adjustments.

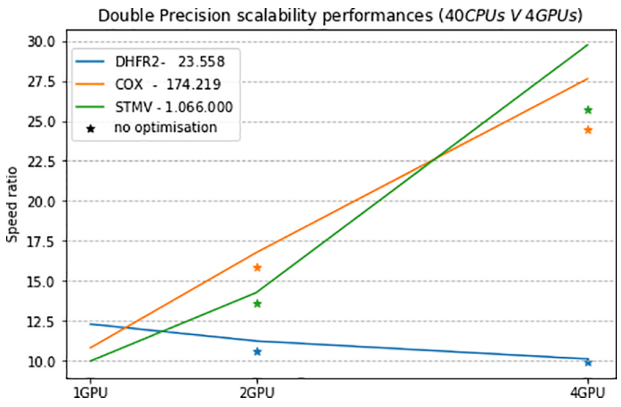


Figure 7. Performance ratio between single node GPU and single node CPU performance. Reference values can be found in Table 1.

Table 1. Single Device Benchmark: MD Production per Day (ns/day)[a]

systems/ devices (ns/day)	CPUs - one node	RTX 2080Ti	RTX 2080Ti + optim	V100	V100 + optim	Tinker-OpenMM V100
DHFR	0.754	2.364	3.903	8.900	9.260	6.300
COX	0.103	0.341	0.563	1.051	1.120	0.957
STMV	0.013	n/a	n/a	0.111	0.126	0.130

[a]All simulations were run using a RESPA/2 fs setup.

In summary, our DP implementation is already satisfactory compared to other applications such as Tinker-OpenMM. Our next section concerns the porting of Tinker-HP in a downgraded precision.

CUDA APPROACH

Even though we already have a robust OpenACC implementation of Tinker-HP in double precision, the gain in terms of computational speed when switching directly to single precision (SP) is modest, as shown in Table 2, which is inconsistent with the GPUs' computational capabilities.

Table 2. Single Precision MD Production (ns/day) within the OpenACC Implementation

	DHFR	COX	STMV
V100	11.69	1.72	0.15
RTX-2080 Ti	11.72	1.51	n/a

This is more obvious for Geforce architecture devices since those cards do not possess DP physical compute units and therefore emulate DP Instructions. According to Table 3, theoretical ratios of 2 and 31 are respectively expected from V100 and RTX-2080 Ti performances when switching from DP to SP, which makes an efficient SP implementation mandatory.

In practice, instead of doubling the speed on V100 cards, we ended up noticing a 1.25 increase factor on V100 and 3 on RTX on DHFR in SP compared to DP with the same setup. All tests have been done under the assumption that our simulations are valid in this precision mode. More results are shown in Table 2. Furthermore, a deep profile conducted on the kernels representing Tinker-HP's bottleneck (real space

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

Table 3. Device Hardware Specifications

GPU	performances (Tflop/s) DP	SP	memory bandwidth (GB/s)	compute/access DP (Ops/8B)	SP (Ops/4B)
Quadro GV100	7.40	14.80	870.0	68.04	68.04
Tesla V100 SXM2	7.80	15.70	900.0	69.33	69.77
Geforce RTX-2080 Ti	0.42	13.45	616.0	5.45	87.33
Geforce RTX-3090	0.556	35.58	936.2	4.75	152.01

nonbonded interactions) in the current state reveals an insufficient exploitation of the GPU SP compute power. Figure 8 suggests that there is still room for improvements in order to take full advantage of the card's computing power and memory bandwidth both in SP and DP. In order to exploit device SP computational power and get rid of the bottleneck exposed by Figure 8, it becomes necessary to reshape our implementation method and consider some technical aspects beyond OpenACC's scope.

Global Overview and Definitions. As mentioned in the previous section, GPUs are most efficient with parallel computations and coalesce memory access patterns. The execution model combines and handles effectively two nested levels of parallelism. The high level concerns multithreading and the low level the SIMD execution model for vectorization.[22,43] This model stands for single instruction multiple threads (SIMT).[44] When it comes to GPU programming, SIMT also includes control-flow instructions along with subroutine calls within the SIMD level. This provides additional freedom of approach during implementation. To improve the results presented in the last paragraph (Table 2) and increase peak performance on computation and throughput, it is crucial to expose more computations in real space kernels and to minimize global memory accesses in order to benefit from cache and shared memory accesses as well as registers. Considering OpenACC paradigm limitations in terms of kernel description as well as the required low-level features, we decided to rewrite those specific kernels using the standard approach of low-level device programming in addition to CUDA built-in intrinsics. In a following section, we will describe our corresponding strategy after a thorough review on precision.

Precision study and Validation.

Definition i. We shall call ϵ_p the machine precision (in SP or DP), the smallest floating point value such that $1 + \epsilon_p > 1$. They are respectively 1.2×10^{-7} and 2.2×10^{-16} in SP and DP.

Definition ii. Considering a positive floating point variable a, the machine precision ϵ_a attached to a is

$$1 + \epsilon > 1 \Leftrightarrow a + \epsilon_p \cdot a > a \Leftrightarrow \epsilon_a = \epsilon_p \cdot a$$

Therefore an error made for a floating point operation between a and b can be expressed as

$$a \tilde{\oplus} b = (a \oplus b)(1 + \epsilon_p) \quad (1)$$

where $\tilde{\oplus}$ designates the numerical operation between a and b.

Property i. Numerical error resulting from sequential reduction operations are linear while those resulting from parallel reduction are logarithmic. Thus, parallel reductions are entirely suitable to GPU implementation as they benefit from both parallelism and accuracy.

Before looking further into the matter of downgrading precision, we have to make sure that Tinker-HP is able to work in this mode. Although it has been proven in the literature[25,27,45] that many MD applications are able to provide correct results with simple precision, extensive precision studies with polarizable force fields are lacking.

When it comes to standard IEEE floating point arithmetic, regular 32 bit storage offers no more than 7 significant digits due to the mantissa. In comparison, we benefit from 16 significant digits with DP 64 storage bits. Without any consideration on the floating number's sign, it is safe to assume that any application working with absolute values outside the $[10^{-7}, 10^{7}]$ scope will fail to deliver sufficient accuracy when operating in complete SP mode. This is called the floating point overflow. To overcome this, the common solution is to use a mixed precision mode (MP) that encompasses both standard SP and a superior precision container to store variables subject to SP overflowing. In practice, most MD applications adopt SP for computation and a higher precision for accumulation. Moreover, applications like Amber or OpenMM propose another accumulation method which rely on a different type of variable.[45]

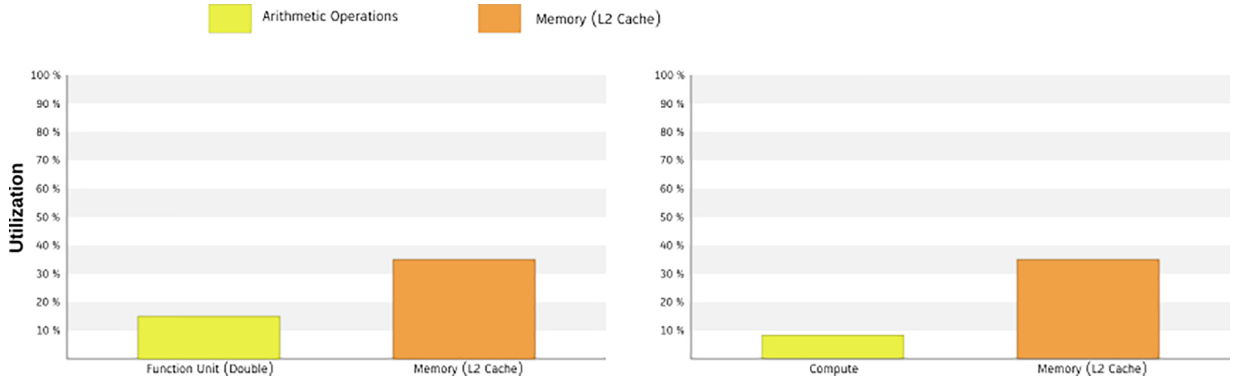


Figure 8. Profile of the matrix-vector compute kernel on the DHFR system. The left picure is obtained with the double precision and the right one with simple precision. In both modes, results indicate an obvious latency issue coming from memory accessing pattern that prevents the device from reaching its peak performance.

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

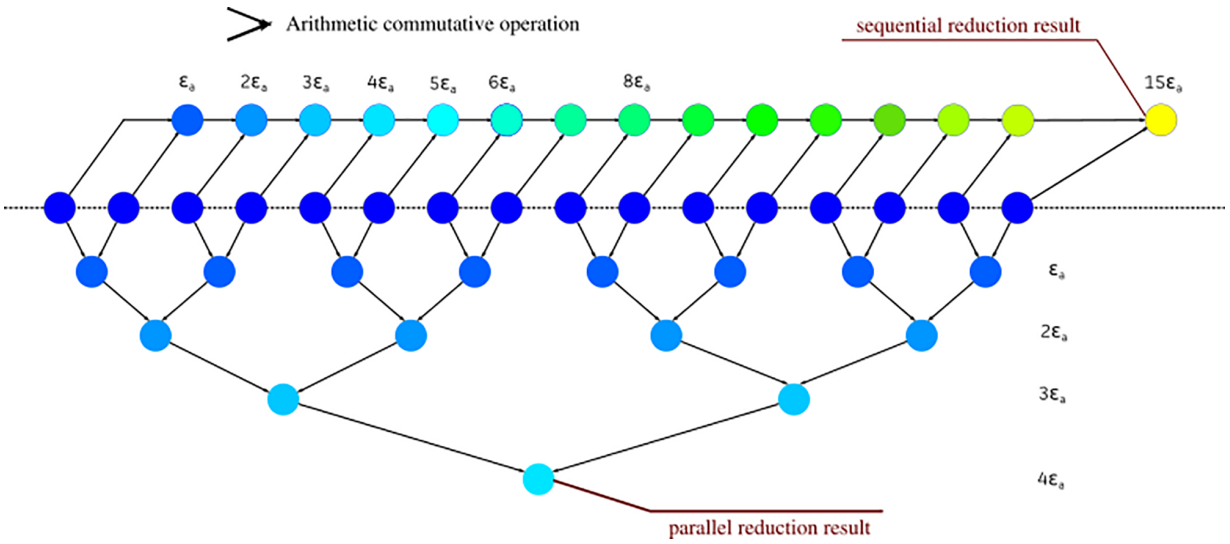


Figure 9. Illustration of the reduction operation on a 16 variables set. Each arithmetic operation generates an error ϵ_a that is accumulated during the sequential operation. On the other hand, parallel reduction uses intermediate variables to significantly reduce the error.

The description made in the previous section shows that energy and the virial evaluation are linear-dependent with the system's size. Depending on the complexity of the interaction in addition to the number of operations it requires, we can associate a constant error value ϵ_i to it. Thus, we can bound the error made on the computation of a potential energy with $N_{\text{int}}\epsilon_i < n\text{Neig}_{\text{max}}\epsilon_i$, where N_{int} represents the number of interactions contributing to this energy and Neig_{max} the maximum number of neighbor it involves per atom. As it is linear with respect to the system size we have to evaluate this entity with a DP storage container. Furthermore, to reduce even more the accumulation of error due to large summation, we shall employ buffered parallel reduction instead of a sequential one (Figure 9). On the other hand, we have to deal with the forces that remain the principal quantities that drive a MD simulation. The error made for each atom on the nonbonded forces is bound by $\text{Neig}_{\text{max}}\cdot\epsilon_i$ depending on the cutoff. However, each potential comes with a different ϵ_i. In practice, the corresponding highest values are the ones of both van der Waals and bonded potentials. The large number of pairwise interactions induced by the larger van der Waals cutoff in addition to the functional form that includes a power of 14 (for AMOEBA) causes SP overflowing for distances greater than 3 Å. By reshaping the variable encompassing the pairwise distance, we get a result much closer to DP since intermediate calculations do not overflow. Regarding the bonded potentials, ϵ_i^{bond} depends more on the conformation of the system.

Parameters involved in bond pairwise evaluation (spring stiffness, ...) cause a SP numerical error (ϵ_i^{bond}) standing between 1×10^{-3} and 1×10^{-2}, which frequently reach 1×10^{-1} (following (eq 1)) during the summation process, and this affects forces more than total energy. In order to minimize ϵ_i^{bond}, we evaluate the distances in DP before casting the result to SP. In the end, ϵ_i^{bond} is reduced on the scope of $[1 \times 10^{-4}, 1 \times 10^{-3}]$, which represents the smallest error we can expect from SP.

Furthermore, unlike the energy, a sequential reduction using atomic operations is applied to the forces. The resulting numerical error is therefore linear with the total number of summation operations. This is why we adopt a 64 bit container for those variables despite the fact they can be held in a 32 bits container.

Regarding the type of the 64 bit container, we analyze two different choices. First, we have the immediate choice of a floating point. The classical mixed precision uses FP64 for accumulation and integration. Every MD applications running on GPU integrates this mode. It presents the advantage of being straightforward to implement. Second, we can use an integer container for accumulation: this is the concept of fixed point arithmetic introduced by Yates.[46] To be able to hold a floating point inside an integer requires us to define a certain number of bits to hold the decimal part. It is called the fixed point fractional bits. The left bits are dedicated to the integer part. Unlike the floating point, freezing the decimal position constrains the approximation precision but offers a correct accuracy in addition to deterministic operations. Considering a floating point value x and an integer one a and a fractional bits value (**fB**), the relations establishing the transition back and forth between them, as $a = f(x)$ and $x = f^{-1}(a)$, are defined as follows:

$$a = \texttt{int}\left(\texttt{round}\left(x \times 2^{\text{fB}}\right)\right) \tag{2}$$

$$x = \frac{\texttt{real}(a)}{2^{\text{fB}}} \tag{3}$$

with `int` and `real` the converting functions and `round` the truncation function that extracts the integer part. When it comes to MD, fixed point arithmetic is an excellent tool: each SP pairwise contribution is small enough to be efficiently captured by a 64 bit fixed point. For instance, it takes only 27 bits to capture 8 digits after the decimal point with a large place left for the integer part. For typical values observed with different system sizes, we are far from the limit imposed by the integer part of the container. Inspired by the work of Walker, Götz, et al.,[45] we have implemented this feature inside Tinker-HP with the following configuration: 34 fractional bits has been selected for forces accumulation, which leaves 30 bits for the integer part, thus setting the absolute limit value to 2^{29} (kcal/mol)·Å. For the energy, we only allocated 30 fractional

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

bits given the fact that it grows linearly with the system size. Besides, using an integer container for accumulation avoids dealing with DP instructions, which significantly affects performance on Geforce cards unlike Tesla ones. In summary, we should expect at least a performance or precision improvement from FP.

A practical verification is shown in Figure 10. In all cases, both MP and FP behave similarly. Forces being the driving

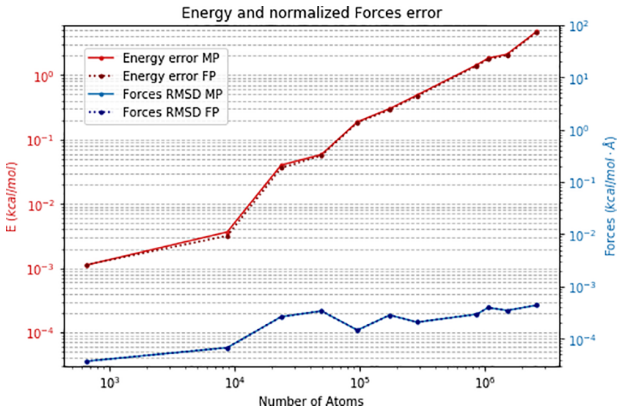


Figure 10. Absolute error between DP implementation and both FP and MP implementations on total potential energy and Forces. Forces root-mean-square deviation between DP and MP for systems from 648 up to 2592000 atoms. As expected, both absolute errors in SP and FP are almost identical on the energy and they grow linearly with the system size. Logarithmic regression gives 0.99 value for the curve slope and up to 5 kcal/mol for the largest system. However, the relative error for all systems is located under 7×10^{-7} in comparison to DP. One can also see that the error on the forces is independent of the system size.

components of MD, the trajectories generated by our mixed precision implementation are accurate. However, as one can see, if errors remain very low for forces even for large systems, a larger error exists for energies, a phenomenon observed in all previous MD GPU implementations. Some specific post-treatment computations, like in a BAR free energy computation or *NPT* simulations with a Monte Carlo barostat, require accurate energies. In such a situation, one could use the DP capabilities of the code for this postprocessing step as Tinker-HP remains exceptionally efficient in DP even for large systems. A further validation simulation in the *NVE* ensemble can be found in Figure 15, confirming the overall excellent stability of the code.

Neighbor List. We want to expose the maximum of computation inside a kernel using the device shared memory. To do so, we consider the approach where a specific group of atoms interacts with another one in a block-matrix pattern (see Figure 11). We need to load the parameters of the group of atoms and the output structures needed for computation directly inside cache memory and/or registers. On top of that, CUDA built-in intrinsics can be used to read data from neighbor threads and if possible compute cross term interactions. Ideally, we can expose $B_{comp} = {B_{size}}^2$ computations without a single access to global memory, with B_{size} representing the number of atoms within the group. With this approach, the kernel should reach its peak in terms of computational load.

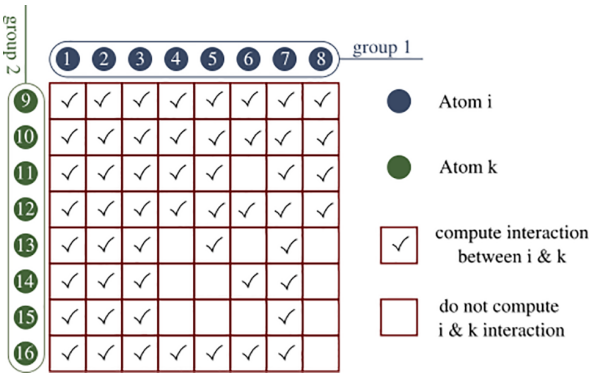


Figure 11. Representation of interactions between two groups of atoms within Tinker-HP. $B_{size} = 8$ for the illustration.

A new approach of the neighbor list algorithm is necessary to follow the logic presented above. This method will be close to standard blocking techniques used in many MD applications.[25,27] Let us present the structure of the algorithm in a sequential and parallel, MPI, context.

Box Partitioning. Lets us recall that given a simulation box $\boldsymbol{\Omega}$, a set of $\boldsymbol{\omega}_c$ with $c \in [0, ..., N_c]$ forms a $\boldsymbol{\Omega}$ partition if and only if

$$\begin{cases} \boldsymbol{\omega}_1 \cup ... \cup \boldsymbol{\omega}_{N_c} = \boldsymbol{\Omega} \\ \boldsymbol{\omega}_1 \cap ... \cap \boldsymbol{\omega}_{N_c} = \varnothing \end{cases}$$

We consider in the following that each group deals with interactions involving atoms within a region of space. In order to maximize B_{comp} between every pair of groups, we must then ensure their spatial compactness. Moreover, all these regions need to define a partition of $\boldsymbol{\Omega}$ to make sure we do not end up with duplicate interactions. Following this reasoning, we might be tempted to group them into small spheres but it is impossible to partition a polygon with only spheres, not to mention the difficulties arising from the implementation point of view.

The MPI layer of Tinker-HP induces a first partition of $\boldsymbol{\Omega}$ in P subdomains $\boldsymbol{\psi}_p$, $p \in [0, ..., P]$, where P is the number of MPI processes. Tinker-HP uses the midpoint image convention[47] so that the interactions computed by the process assigned to $\boldsymbol{\psi}_p$ are the ones whose midpoint falls into $\boldsymbol{\psi}_p$. The approach used in Tinker-HP for the nonbonded neighbor list uses a cubic partition $\boldsymbol{\omega}_c$, $c \in [1, ..., N_c]$, of $\boldsymbol{\psi}_p$ and then collects the neighboring atoms among the neighboring cells of $\boldsymbol{\omega}_c$. Here, we proceed exactly in the same way with two additional conditions to the partitioning. First, the number of atoms inside each cell $\boldsymbol{\omega}_c$ must be less or equal than B_{size}. Second, we must preserve a common global numbering of the cells across all domains $\boldsymbol{\psi}_p$ to benefit from a unique partitioning of $\boldsymbol{\Omega}$.

Once the first partitioning in cells is done, an additional sorting operation is initiated to define groups so that each of them contains exactly B_{size} spatially aligned atoms following the cell numbering (note that because of the first constrain mention earlier, one cell can contain atoms belonging to a maximum of two groups). More precisely, the numbering of the cells follows a one-dimensional representation of the three dimension of the simulation box. Now, we want to find the best partitioning of $\boldsymbol{\psi}_p$ in groups that will ensure enough proximity between atoms inside a group, minimizing the

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

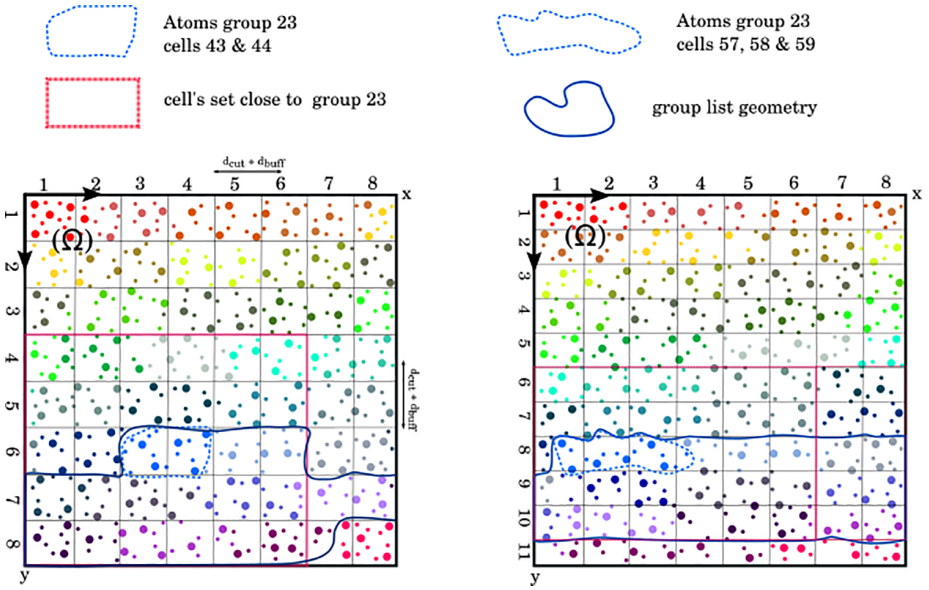


Figure 12. Illustration of a two-dimensional partition along with groups for a box of water. The left figure shows a 64-cell partition of **Ω** while the right one refines this partitioning into 88 cells. The groups are defined by reindexing the atoms following the cell numbering and their maximum size. Here $B_{size} = 16$. A unique color is associated with every atom belonging to the same group. No cell contains more than B_{size} atoms or 2 groups. Once a group is selected (23), searching for its neighboring groups is made through the set of cells (with respect to the periodic boundary conditions) near the cells it contains (43 and 44). Once this set is acquired, all the group indexes greater than or equal to the selected group constitute the actual list of neighbors to take the symmetry of the interactions into account. We see that the group's shape modulates the group neighborhood as illustrated with the right illustration and a spatially flat group 23.

number of neighboring groups and consequently maximizing B_{comp}.

When the partitioning generates too flat domains, each group might end up having too many neighboring groups. The optimal cell shape (close to a sphere) is the cube but we must not forget the first constraint and end up with a very thin partition either. However, atom groups are not affected by a partition along the innermost contiguous dimension in the cell numbering. We can exploit this to get better partitioning. Figure 12 illustrates and explains the scheme on a two-dimensional box. Partitioning is done in an iterative manner by cycling on every dimension. We progressively increase the number of cells along each dimension starting on the contiguous one until the first condition is fulfilled. During a parallel run, we keep track of the cell with the smallest number of atoms with a reduction operation. This allows to have a global partitioning of **Ω** and not just $\boldsymbol{\psi}_p$.

Now that we do dispose of a spatial rearrangement of the atoms into groups, we need to construct pair-lists of all interacting groups according to the cutoff distance plus an additional buffer to avoid reconstructing it at each time step.

Groups are built in such a way that it is straightforward to jump from groups indexing to cells indexing. We chose to use an adjacency matrix which is GPU suitable and compatible with MPI parallelism.

Once it is built, the adjacency matrix directly gives the pair-list. Regarding the storage size involved with this approach, note that we only require single bit to tag pair−group interactions. This results in an $\left\lceil \frac{\mathbf{n}_l}{B_{size}} \right\rceil^2$ bits occupation that equals to $\left\lceil \frac{n}{B_{size}} \right\rceil^2 \frac{1}{8}$ bytes. $\boldsymbol{n}_l$ represents the number of atoms which participates to real space evaluation on a process domain ($\boldsymbol{\psi}_p$). Of course, in terms of memory we cannot afford a quadratic reservation. However, the scaling factor $\left\lceil \frac{1}{B_{size}} \right\rceil^2 \frac{1}{8}$ is small enough even for the smallest value of B_{size} set to 32 corresponding to device warp size. Not to mention that, in the context of multidevice simulation, the memory distribution is also quadratic. The pseudokernel is presented in Chart 3.

Once the adjacency matrix is built, a simple postprocessing gives us the adjacency list with optimal memory size and we can use the new list on real space computation kernels following the process described in the introduction of this subsection and illustrated in Figure 11. In addition, we benefit from a coalesced memory access pattern while loading blocks data and parameters when they are spatially reordered.

List Filtering. It is possible to improve the performance of the group−group pairing with a similar approach to the list reordering method mentioned in the OpenACC optimizations section above. By filtering every neighboring group, we can get a list of atoms that really belong to a group's neighborhood. The process is achieved by following the rule:

$$\alpha \in \mathcal{B}_I \quad \text{if } \exists \alpha_i \in \beta_I \quad \text{such that } \mathsf{dist}(\alpha_i, \alpha) \leq \mathrm{d_{cut}} + \mathrm{d_{buff}}.$$

α and α_i are atoms, β represents a group of B_{size} atoms, $\mathcal{B}$ is the neighborhood of a group and $\mathtt{dist}: (\mathbb{R}^3 \times \mathbb{R}^3) \rightarrow \mathbb{R}$ is the euclidean distance.

An illustration of the results using the filtering process is depicted in Figure 13.

When the number of neighbor atoms is not a multiple of B_{size}, we create phantom atoms to complete the actual neighbor lists. A drawback of the filtering process is a loss of coalesced memory access pattern. As it has been entirely constructed in parallel, we do not have control of the output order. Nonetheless, this is compensated by an increase of B_{comp} for each interaction between groups, as represented by Figure 13.

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

Chart 3. Adjacency Matrix Construction Pseudo-kernel[a]

```
c$acc parallel loop default(present)
do i = 1, numCells
   celli = i
   !get blocks_i inside celli
   ...
c$acc loop vector
   do j = 1, numCellsNeigh
      ! Get cellj with number
      ...
      ! Get blocks_j inside cellk
      ...
      ! Apply symmetrical condition
      if ( cellj > celli ) cycle
c$acc loop seq
      do bi in blocks_i
         do bj in blocks_j
c$acc atomic
            set matrix(bj,bi) to 1
         end do
      end do
   end do
end do
```

[a]We browse through all the cells, and for each one we loop on their neighbors. It is easy to compute their ids since we know their length as well as their arrangement. Given the fact that all cells form a partition of the box, we can apply the symmetrical condition on pair-cells and retrieve the groups inside thanks to the partitioning condition, which ensures that each cell contains at most two groups.

In practice, we measure a 75% performance gain between the original list and the filtered one for the van der Waals interaction kernel. Moreover, Figure 14 (deep profile of the previous bottleneck kernel: matrix-vector product) shows a much better utilization of the device computational capability. We apply the same strategy for the other real space kernels (electrostatics and polarization).

PME Separation. As mentioned above, the Particle Mesh Ewald method separates electrostatics computation in two, real and reciprocal space. A new profiling of Tinker-HP in single-device mixed precision mode with the latest developments shows that the reciprocal part is the new bottleneck. More precisely, real space performs 20% faster than reciprocal space within a standard PME setup. Moreover, reciprocal space is even more a bottleneck in parallel because of the additional MPI communications induced by the cuFFt Transformations. This significantly narrows our chances of benefiting from the optimizations mentioned in the previous optimization subsection. However, as both parts are independent, we can distribute them on different MPI processes in order to reduce or even suppress communications inside FFts. During this operation, a subset of GPUs are assigned to reciprocal space computation only. Depending on the system size and the load balancing between real and reciprocal spaces, we can break through the scalability limit and gain additional performance on a multidevice configuration.

Mixed Precision Validation. To validate the precision study made above, we compare a 1 ns long simulation in DP on CPU (Tinker-HP 1.2) in a constant energy setup (*NVE*) with the exact same run using both GPU MP and FP implementations.

We used the solvated DHFR protein and the standard velocity verlet integrator with a 0.5 fs time step, 12 and 7 Å cutoff distances respectively for van der Waals and real space electrostatics and a convergence criteria of 1×10^{-6} for the polarization solver. A grid of $64 \times 64 \times 64$ was used for reciprocal space with fifth-order splines. We also compare our results with a trajectory obtained with Tinker-OpenMM in MP in the exact same setup; see Figure 15.

The energy is remarkably conserved along the trajectories obtained with Tinker-HP in all cases: using DP, MP or FP with less oscillations than with Tinker-OpenMM with MP.

Available Features. The main features of Tinker-HP have been offloaded to GPU such as its various integrators like the multi-time-step integrators: RESPA1 and BAOAB-RESPA1,[48] which allow up to a 10 fs time step with PFF (this required to create new neighbor lists to perform short-range nonbonded interactions computations for both van der Waals and electrostatics). Aside from Langevin integrators, we ported

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

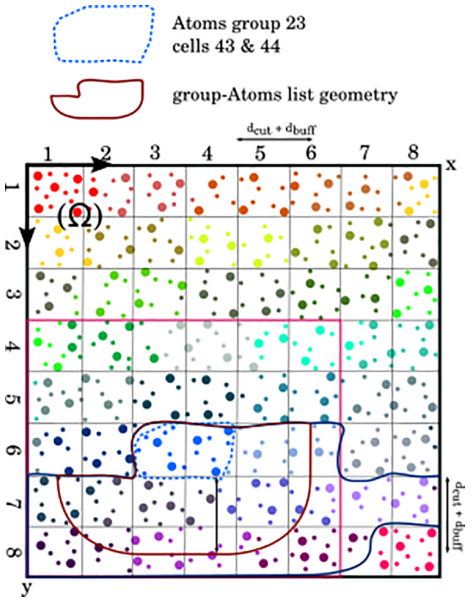


Figure 13. Starting from the situation illustrated by Figure 12, we represent the geometry resulting from the filtering process. We significantly reduce the group 23 neighborhood with the list filtering; it decreases from 144 atoms with the first list to 77 with the filtered one. B_{comp} increases, which corresponds to more interactions computed within each group pair.

the Bussi[42] (which is the default) and the Berendsen thermostats, as well as the Monte Carlo and the Berendsen barostats. We also ported free energy methods such as the Steered Molecular Dynamics[49] and van der Waals soft cores for alchemical transformations, as well as the enhanced sampling method Gaussian Accelerated Molecular Dynamics.[50] Even if it is not the main goal of our implementation as well optimized software suited to such simulations exist, we also ported the routines necessary to use standard nonpolarizable force fields such as CHARMM,[4] Amber,[5] or OPLS.[51] Still, we obtained already satisfactory performances with these models despite a simple portage, the associated numbers can be found in the Supporting Information and further optimization is ongoing. On top of all these features that concern a molecular dynamics simulation, we ported the "analyze" and "minimize" program of Tinker-HP, allowing to run single point calculations as well as geometry optimizations. All these capabilities are summed up in Table 4.

Performance and Scalability Results. We ran benchmarks with various systems on a set of different GPUs in addition to Tesla V100 nodes of the Jean-Zay supercomputer. We also ran the whole set of tests on the Irène Joliot Curie ATOS Sequana supercomputer V100 partition to ensure for the portability of the code. We used two different integrators: (2 fs RESPA along with 10 fs BAOAB-RESPA1 with heavy hydrogens). For each system, we performed 2.5 and 25 ps MD simulations with RESPA and BAOAB-RESPA1, respectively, and averaged the performance on the complete runs. van der Waals and real space electrostatics cutoffs were respectively set to 9 and 7 Å plus 0.7 Å neighbor list buffer for RESPA, 1 Å for BAOAB-RESPA1. We used the Bussi thermostat with the RESPA integrator. Induced dipoles were converged up to a 1×10^{-5} convergence threshold with the conjugate gradient solver and a diagonal preconditioner.[37] The test cases are water boxes within the range of 96000 atoms (i.e., `Puddle`) up to 2592000 atoms (i.e., `Bay`), the `DHFR`, `COX` and the Main Protease of Sars-Cov2 proteins (M^{pro})[52] as well as the `STMV` virus. Table 5 gathers all single devices performances, and Figure 16 illustrates the multidevice performance.

On a single GPU, the BAOAB-RESPA1 integrator performs almost twice as fast as RESPA in all cases: 22.53−42.83 ns/day on `DHFR`, 0.57−1.11 ns/day for the `STMV` virus. Regarding the RESPA integrator, results compared with those obtained in DP (Table 1) are now consistent with the Quadro V100 theoretical performance. Moreover, we observe a significant improvement on single V100 cards with DP in comparison to the OpenACC implementation, which shows that the algorithm is better suited to the architecture. However, this new algorithm considerably underperforms on Geforce architecture. For instance, for the `COX` system the speed goes from 0.65 ns/j with the OpenACC implementation to 0.19 ns/j with the adapted CUDA implementation on Geforce RTX-2080 Ti. This is obviously related to architecture constraints (lack of DP Compute units, sensitivity to SIMD divergence branch, instruction latency) and shows that there is still room for optimization. Tinker-HP is tuned to select the quickest algorithm depending on the target device. Concerning MP performance on Geforce Cards, we finally get the expected ratio compared with DP: increasing computation per access improves the use of the device (Table 3). Geforce RTX-2080 Ti and GV100 results are close until the `COX` test case, which

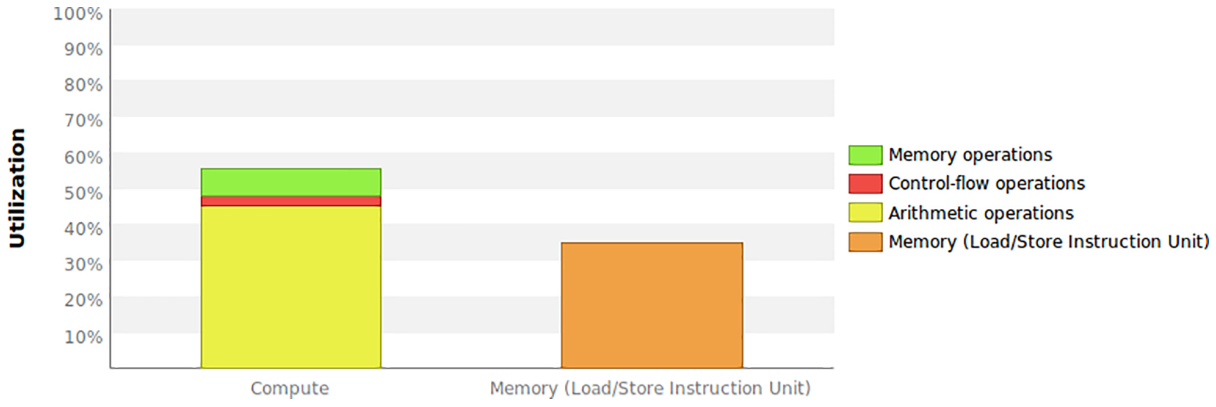


Figure 14. Real space kernel profiling results in mixed precision using our new group-Atoms list.

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

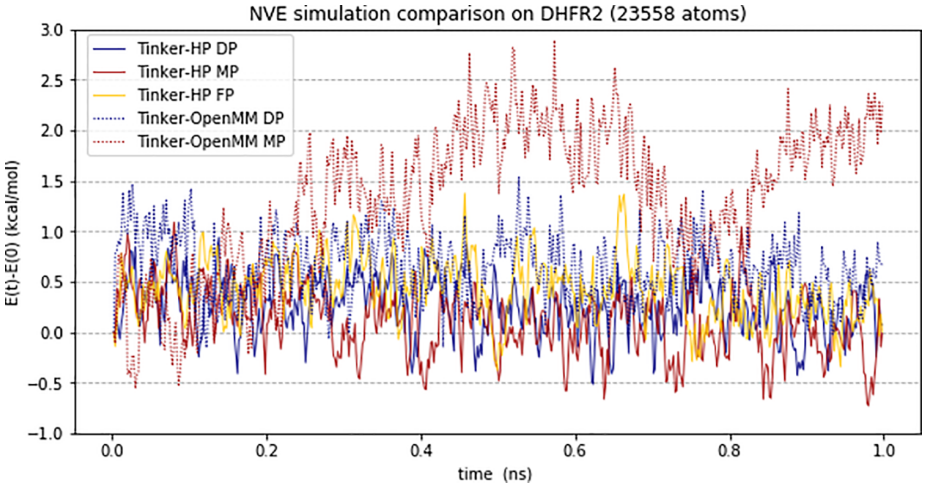


Figure 15. Variation of the total energy during a *NVE* molecular dynamics simulation of the DHFR protein in DP and MP and FP. Energy fluctuations are respectively within 1.45, 1.82, 1.75, 1.69, and 3.45 kcal/mol for Tinker-HP DP SP FP and Tinker-OpenMM DP MP.

Table 4. Available Features in the Initial Tinker-HP GPU Release

programs	dynamic; analyze; minimize; bar
integrator	VERLET (default); RESPA; RESPA1; BAOAB-RESPA; BAOAB-RESPA1
force fields	AMOEBA; CHARMM/AMBER/OPLS
miscellaneous	steered MD (SMD); Gaussian accelerated MD; restrained groups; soft cores; plumed
thermostat	Bussi (default); Berendsen
barostat	Berendsen (default); Monte Carlo

Table 5. Tinker-HP Performances in (ns/day) on Different Devices and Precision Modes

	systems						
	DHFR	M^{pro}	COX	Pond	Lake	STMV	Bay
DP Quadro GV100							
RESPA 2 fs	11.24	2.91	1.76	1.08	0.36	0.24	0.11
BAOAB-RESPA1 10 fs	22.03	6.09	3.61	2.25	0.76	0.53	0.24
MP							
RESPA 2 fs	21.75	5.98	3.69	2.20	0.70	0.44	0.20
BAOAB-RESPA1 10 fs	40.73	12.80	3.61	4.58	1.49	1.01	0.46
FP							
RESPA 2 fs	21.46	5.82	3.57	2.12	0.67	0.43	0.20
BAOAB-RESPA1 10 fs	40.65	12.65	7.77	4.52	1.47	1.00	0.45
MP Geforce RTX-2080 Ti							
RESPA 2 fs	22.52	5.35	3.21	1.82	0.54	0.33	0.15
BAOAB-RESPA1 10 fs	43.81	11.85	7.06	4.06	1.24	0.82	n/a
FP							
RESPA 2 fs	24.95	5.73	3.45	1.95	0.57	0.35	0.16
BAOAB-RESPA1 10 fs	47.31	12.78	7.63	4.35	1.32	0.87	n/a
MP Geforce RTX-3090							
RESPA 2 fs	29.14	7.79	4.76	2.81	0.91	0.60	0.28
BAOAB-RESPA1 10 fs	52.80	15.79	9.61	5.52	1.81	1.23	0.59
FP							
RESPA 2 fs	32.00	8.37	5.10	3.02	0.96	0.64	0.30
BAOAB-RESPA1 10 fs	57.67	17.20	10.46	5.96	1.90	1.32	0.63

is consistent with their computing power, but GV100 performs better for larger systems. It is certainly due to the difference in memory bandwidth which allows GV100 to perform better on memory bound kernels and to reach peak performance more easily. For example, most of PME reciprocal space kernels are memory bounded due to numerous accesses to the three-dimensional grid during the building and extracting process.

A further comparison between architectures is given in the Supporting Information.

For FP simulations, as expected, we do not see any performance difference with MP on V100 cards unlike Geforce ones, which exhibit an 8% acceleration in average as the DP accumulation is being replaced by an integer one (an instruction natively handled by compute cores). Table 6 shows the performance of Tinker-OpenMM: with the same RESPA framework, Tinker-HP performs 12−30% better on GV100 when the system size grows. With Geforce RTX-2080 Ti the difference is slightly more steady except for the Lake test case: around 18% and 25% better performance with Tinker-HP respectively with MP and FP compared to Tinker-OpenMM.

The parallel scalability starts to be effective above 100000 atoms. This is partly because of the mandatory host synchronizations needed by MPI and because of the difference in performance between synchronous and asynchronous computation under that scale (for example, DHFR production drops to 12 ns/day when running synchronously with the host). Kernel launching times are almost equivalent to their execution time and they do not overlap. Each GPU on the Jean-Zay Supercomputer comes with a 300 GB/s interconnection NVlink bandwidth. Four GPUs per node, all of them being interconnected, represents then a 100 Gb/s interconnection for each GPU pairs. The third generation PCI-Express bridge to the host memory only delivers 16 Gb/s. With the RESPA integrator operating on a full node made of 4 T V100, the speed ratio grows from 1.14 to 1.95, respectively, from Puddle to Bay test cases in comparison to a single device execution. The relatively balanced load between PME real and reciprocal space allows us to break through the scalability limit on almost every run with 2 GPUs with PME separation enabled. Performance is always worse on 4 GPUs with 1 GPU dedicated to the reciprocal space and the others to the direct space for the same reason mentioned earlier (direct/reciprocal

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

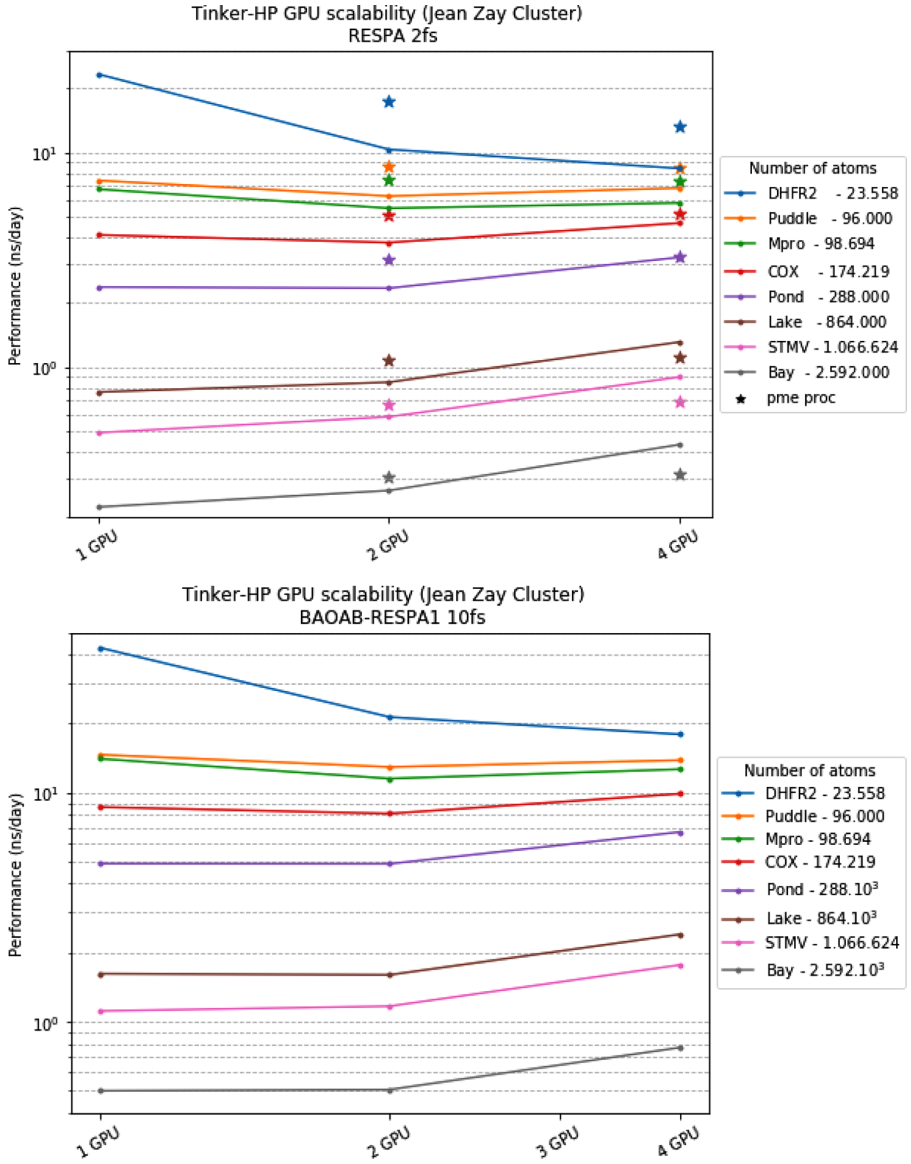


Figure 16. Single node mixed precision scalability on the Jean-Zay Cluster (V100) using the AMOEBA polarizable force field.

Table 6. Tinker-OpenMM Mixed Precision Performances Assessed with the RESPA Framework

	systems						
	DHFR	M^{pro}	COX	Pond	Lake	STMV	Bay
Quadro GV100	17.53	4.50	2.56	1.68	0.56	0.34	n/a
Geforce RTX-2080 Ti	18.97	4.37	2.63	1.66	0.55	0.28	n/a

space load balancing). We also diminished the communication overhead by overlapping communication and computation. Note that on a complete node of Jean-Zay with 4 GPUs, the bandwidth is statically shared between all of them, which means that the performance showed here on 2 GPUs is less that what can be expected on a node that would only consist in 2 GPUs interconnected through NVlink. With the BAOAB-RESPA1 integrator, ratios between a full node and a single device vary from 1.07 to a maximum of 1.58. Because of the additional short-range real space interactions, it is unsuited for PME separation, yet the reduced amount of FFt offers a potential for scalability higher than RESPA. Such a delay in the strong scalability is understandable given the device computational speed, the size of the messages size imposed by the parallel distribution, and the configuration run. The overhead of the MPI layer for STMV with BAOAB-RESPA1 and a 4 GPU bench is on average 41% of a time step. It consists mostly in FFt grid exchange in addition to the communication of dipoles in the polarization solver. This is an indication of the theoretical gain we can obtain with an improvement of the interconnect technology or the MPI layer. Ideally, we can expect to produce 2.63 ns/day on a single node instead of 1.55 ns/day. It is already satisfactory to be able to scale on such huge systems and further efforts will be made to improve multi-GPU results in the future.

TOWARD LARGER SYSTEMS

As one of the goals of the development of Tinker-HP is to be able to treat (very) large biological systems such as protein

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

complexes or entire viruses encompassing up to several millions of atoms (as it is already the case with the CPU implementation[20,22] by using thousands of CPU cores), we review in the following section the scalability limit of the GPU implementation in terms of system size knowing that GPUs do not have the same memory capabilities: where classical CPU nodes routinely benefit from more than 128 GB of memory, the most advanced Ampere GPU architecture holds up to 40 GB of memory.

Tinker-HP Memory Management Model. MD with 3D spatial decomposition has its own pattern when it comes to memory distribution among MPI processes. We use the midpoint rule to compute real space interactions as it is done in the CPU implementation.

In practice, it means that each process holds information about its neighbors (to be able to compute the proper forces). More precisely, a domain $\boldsymbol{\psi}_q$ belongs to the neighborhood of $\boldsymbol{\psi}_p$ if the minimum distance between them is under some cutoff distance plus a buffer. To simplify data exchange between processes, we transfer all positions in a single message; the same thing is done with the forces.

An additional filtering is then performed to list the atoms actually involved in the interactions computed by a domain $\boldsymbol{\psi}_p$. An atom, $\alpha \in \boldsymbol{\Omega}$, belongs to domain $\boldsymbol{\psi}_p$'s interaction area ($\boldsymbol{\lambda}_p$) if the distance between this atom and the domain is below $\frac{d_{\text{cut}} + d_{\text{buff}}}{2}$.

Let us call $\boldsymbol{n}_p$ the number of atoms belonging to $\boldsymbol{\psi}_p$, $\boldsymbol{n}_b$ the number of atoms belonging to a process domain and its neighbors, and $\boldsymbol{n}_l$ the number of atoms inside $\boldsymbol{\lambda}_p$. This is illustrated in Figure 17.

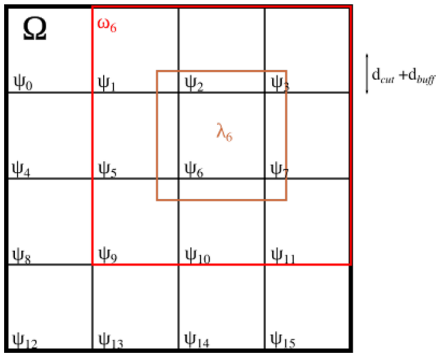


Figure 17. Two-dimensional spatial decomposition of a simulation box with MPI distribution across 16 processes. ω_6 collects all the neighboring domains of $\boldsymbol{\psi}_6$. Here $\boldsymbol{n}_b < \boldsymbol{n}$.

One can see that all data reserved with a size proportional to $\boldsymbol{n}_p$ are equally distributed among processes. Those with size proportional to $\boldsymbol{n}_b$ are only partially distributed. This means that these data structures are not distributed if all domains $\boldsymbol{\psi}_p$ are neighbors. This is why in practice the distribution only takes place at that level with a relatively high number of process, more than 26 at least on a large box with 3d domain decomposition. On the other hand, data allocated with a size proportional to $\boldsymbol{n}_l$ (like the neighbor list) are always more distributed when the number of processes increases.

On top of that, some data remain undistributed (proportional to $\boldsymbol{n}$) like the atomic parameters of each potential energy term. Splitting those among MPI processes would severely increase the communication cost, which we can not afford. As we cannot predict how one atom will interact and move inside $\boldsymbol{\Omega}$, the best strategy regarding such data is to make it available to each process. Reference Tinker-HP reduces the associated memory footprint by using MPI shared memory space: only one parameter data instance is shared among all processes within the same node.

No physical shared memory exists between GPUs of a node, and the only way to deal with undistributed data is by replicating them on each device, which is quickly impractical for large systems.

In the next section, we detail a strategy allowing to circumvent this limitation.

NVSHMEM Feature Implementation. As explained above, distribution of parameter data would necessarily results in additional communications. Regarding data exchange optimizations between GPU devices, NVIDIA develops a new library based on the OpenSHMEM[53] programming pattern, which is called NVSHMEM.[29] This library provides thread communication routines that operate on a symmetric memory on each device meaning that it is possible to initiate device communication inside kernels and not outside with an API like MPI. The immediate benefit of such approach resides in the fact that communications are automatically recovered by kernel instructions and can thereby participate to recover device internal latency. This library allows us to distribute $\boldsymbol{n}$ scale data over devices within one node.

Our implementation follows this scheme: divide a data structure (an array for instance) across devices belonging to the same node following the global numbering of the atoms and access this data inside a kernel with the help of NVSHMEM library. To do that, we rely on a NVHSMEM feature that consists of storing a symmetric memory allocation address in a pointer on which arithmetic operations can be done. Then, depending on the address returned by the pointer, either a global memory access (GBA) or a remote memory access (RMA) is instructed to fetch the data. The implementation requires a Fortran interface to be operational since NVSHMEM source code is written in the C language. Moreover, an additional operation is required for every allocation performed by the NVSHMEM specific allocator to make the data allocated accessible through OpenACC kernels. See Figure 18.

Such a singular approach affects performances since additional communications have to be made inside kernels. Furthermore, all communications do not follow a special pattern that would leave room for optimizations, meaning that each device accesses data randomly from the others depending on the atoms involved in the interactions it needs to compute. In order to limit performance loss, we can decide which data are going to be split across devices and which kernels are going to be involved with this approach. In practice, we use this scheme for the parameters of the bonded potentials.

Doing so, we distribute most of the parameter data (torsions, angles, bonds, ...) and therefore reduce the duplicated memory footprint.

Perspectives and Additional Results. During our NVSHMEM implementation, we were able to detect and optimize several memory wells. For instance, the adjacency matrix described in the Neighbor List subsection has a quadratic memory requirement following the groups of atoms. This means that this represents a potential risk of memory saturation on a single device. To prevent this, we implemented a buffer limit on this matrix to construct the pair-group list

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

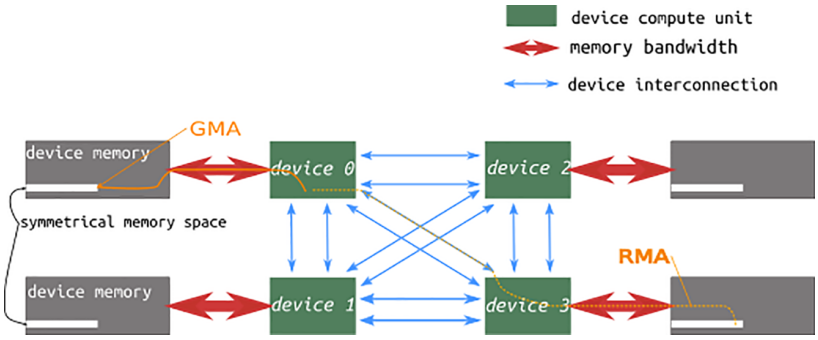


Figure 18. NVSHMEM memory distribution pattern across a four interconnected devices node. A symmetric reserved space is allocated by the NVSHMEM library at initialization. Thus, data are equally split across all devices in order. Every time a device needs to access data allocated with NVSHMEM, either a GMA or RMA is issued.

piece by piece. We also implemented algorithms that prioritize computing and searching over storing where ever needed, essentially scaling factor reconstruction. In the end, Tinker-HP is able to reach a performance of 0.15 ns/day for a 7776000 atoms water box with the AMOEBA force field and the BAOAB-RESPA1 integrator on a single V100 and scale-out to 0.25 ns/day on a complete node of Jean-Zay on the same system.

We also had the opportunity to test our implementation on the latest generation NVIDIA GPU Ampere architecture: the Selene supercomputer which is made of nodes consisting in DGX-A100 servers. A DGX-A100 server contains eight A100 graphic cards with 40 GB of memory each and with latest generation interconnection NVIDIA Switches. The results we obtained on such a node with the same systems as above in the same RESPA and BAOAB-RESPA1 framework are listed in Table 7 and Figure 19.

Table 7. Performance Synthesis and Scalability Results on the Jean-Zay (V100) and Selene (A100) Machines[a]

		Jean-Zay (V100)		Selene (A100)	
systems	size (no. of atoms)	perf (ns/day); 1GPU	best perf (ns/day); #GPU	perf (ns/day); 1GPU	best perf (ns/day); #GPU
DHFR	23558	43.83	43.83	44.96	44.96
Puddle	96000	14.63	15.76; 4	15.57	17.57
Mpro	98694	14.03	14.57; 4	16.36	17.47; 4
COX	174219	8.64	10.15; 4	10.47	11.75; 4
Pond	288000	4.90	6.72; 4	6.18	10.60; 8
Lake	864000	1.62	2.40; 4	2.11	5.50; 8
STMV	1066624	1.11	1.77; 4	1.50	4.51; 8
SARS-Cov2 Spike-ACE2	1509506	0.89	1.55; 4	1.32	4.16; 8
Bay	2592000	0.50	0.77; 4	0.59	2.38; 8
Sea	7776000	0.15	0.25; 4	0.22	0.78; 8

[a]MD production in ns/day with the AMOEBA polarizable force field.

We observe an average of 50% of performance gain for systems larger than 100000 atoms on a single A100 compared to a single V100 card. Also, the more efficient interconnection between cards (NV-switch compared to NV-link) allows us to scale better on several GPUs with the best performances ever obtained with our code on all the benchmark systems, the larger ones making use of all the 8 cards of the node. Although the code is designed to do so, the latency and the speed of the internode interconnection on the present Jean-Zay and Selene supercomputers did not allow us to scale efficiently across nodes, even on the largest systems. Jean-Zay provides 32 GB/s of network interconnection between nodes so that each GPU pair has access to a 16 GB/s bandwidth. Unlike the 100 GB/s shared between each GPU inside a node, we expect internode transit times to be 6.25−12.5 time slower without taking the latency into account. This is illustrated by the experiment summarized in Table 8 as we observe the sudden increase of the overhead of the MPI layer relative to the total duration of a time step when running on two nodes. In this case, changing the domain decomposition dimension to limit the number of neighboring process quadruples the production and exposes the latency issue (expressed here by the difference between the fastest and the slowest MPI process). In a multinode context the bottleneck clearly lies in the internode communications. The very fast evolution of the compilers, as well as the incoming availability of new classes of large pre-exascale supercomputers may improve this situation in the future. Presently, the use of multiple nodes for a single trajectory is the subject of active work within our group and results will be shared in due course. Still, one can already make use of several nodes with the present implementation by using methods such as unsupervised adaptive sampling as we recently proposed.[52] Such pleasingly parallel approach already offers the possibility to use hundreds (if not thousands!) of GPU cards simultaneously.

CONCLUSION

We presented the native Tinker-HP multi-GPU multiprecision acceleration platform. The new code is shown to be accurate and scalable across multiple GPU cards offering unprecedented performances and new capabilities to deal with long time scale simulations on large realistic systems using polarizable force fields such as AMOEBA. The approach strongly reduces the time to solution offering to achieve routine simulations that would have required thousands of CPUs on a single GPU card. Overall, the GPU-accelerated Tinker-HP reaches the best performances ever obtained for AMOEBA simulations and extends the applicability of polarizable force fields. The package is shown to be compatible with various computer GPU system architectures ranging from research laboratories to modern supercomputers.

Future work will focus on adding new features (sampling methods, integrators, ...) and on further optimizing the performance on multinodes/multi-GPUs to address the exascale challenge. We will improve the nonpolarizable force

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

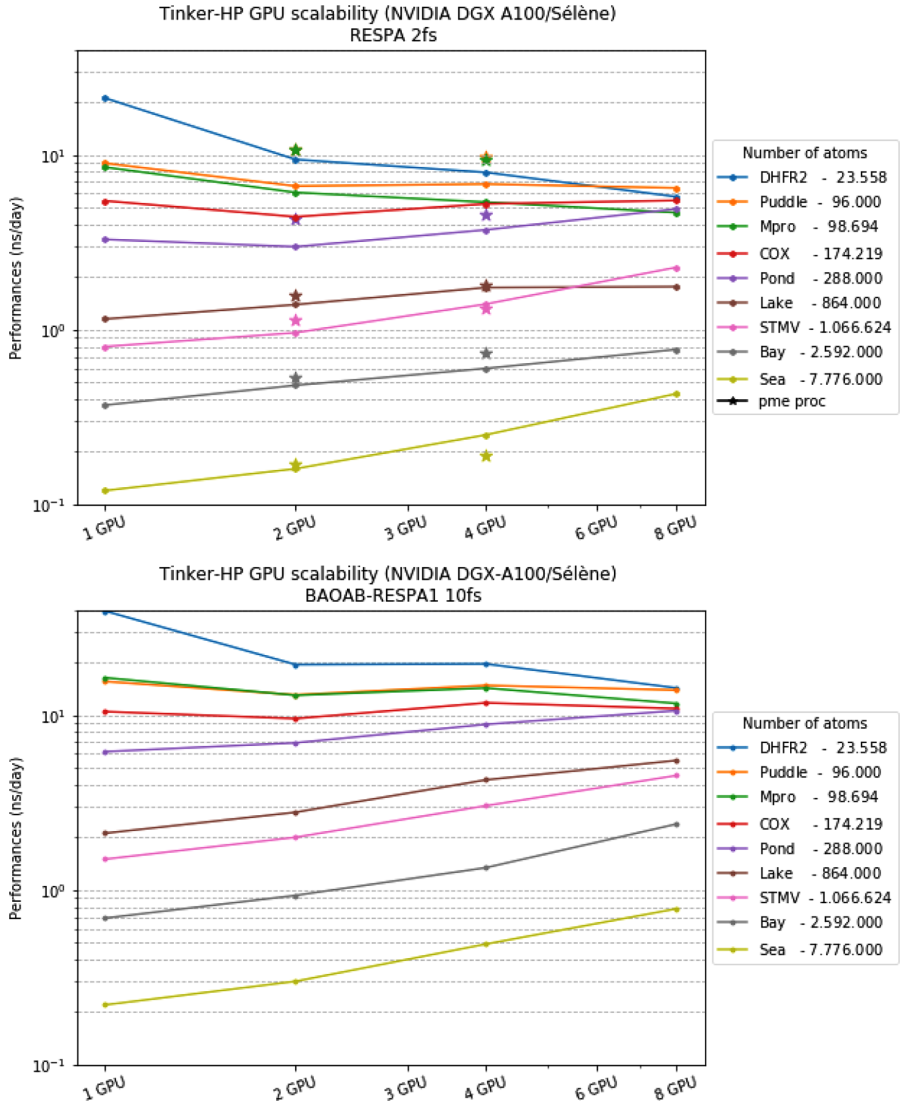


Figure 19. Performance and one node scalibility results with the AMOEBA force field.

Table 8. Multinode Performance on Jean-Zay with the Sea System and BAOAB-RESPA1[a]

no. of GPU:	4		8	
domain decomposition:	3d	1d	3d	1d
production speed (ns/day)	0.252	0.265	0.02	0.08
MPI layer (%)	24	22	97	91
MPI latency (%)	1	1	4	11

[a]Here the latency designates the time difference between the fastest and the slowest process.

field simulations capabilities as we will provide the high performance implementations of additional new generation polarizable many-body force fields such as AMOEBA+,[54,55] SIBFA,[56] and others. We will continue to develop the recently introduced adaptive sampling computing strategy enabling the simultaneous use of hundreds (thousands) of GPU cards to further reduce time to solution and deeper explore conformational spaces at high-resolution.[52] With such exascale-ready simulation setup, computations that would have taken years can now be achieved in days thanks to GPUs. Beyond this native Tinker-HP GPU platform and its various capabilities, an interface to the Plumed library[57] providing additional methodologies for enhanced-sampling, free-energy calculations, and the analysis of molecular dynamics simulations is also available. Finally, the present work, which extensively exploits low precision arithmetic, highlights the key fact that high-performance computing (HPC) grounded applications such as Tinker-HP can now efficiently use converged GPU-accelerated supercomputers, combining HPC and artificial intelligence (AI) such as the Jean-Zay machine to actually enhance their performances.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.0c01164.

Additional information regarding the performance using nonpolarizable force fields as well as a comparison between peak performance reachable by Tinker-HP in terms of FLOP/s (PDF)

https://doi.org/10.1021/acs.jctc.0c01164
J. Chem. Theory Comput. 2021, 17, 2034−2053

AUTHOR INFORMATION

Corresponding Authors

Louis Lagardère − *Sorbonne Université, F-75005 Paris, France; Sorbonne Université, F-75005 Paris, France*; Email: louis.lagardere@sorbonne-universite.fr

Jean-Philip Piquemal − *Sorbonne Université, F-75005 Paris, France; Department of Biomedical Engineering, The University of Texas at Austin, Austin, Texas 78712, United States*; orcid.org/0000-0001-6615-9426; Email: jean-philip.piquemal@sorbonne-universite.fr

Authors

Olivier Adjoua − *Sorbonne Université, F-75005 Paris, France*

Luc-Henri Jolly − *Sorbonne Université, F-75005 Paris, France*

Arnaud Durocher − *Eolen, 75116 Paris, France*

Thibaut Very − *IDRIS, CNRS, 91403 Orsay, France*

Isabelle Dupays − *IDRIS, CNRS, 91403 Orsay, France*

Zhi Wang − *Department of Chemistry, Washington University in Saint Louis, Saint Louis, Missouri 63110, United States*

Théo Jaffrelot Inizan − *Sorbonne Université, F-75005 Paris, France*

Frédéric Célerse − *Sorbonne Université, F-75005 Paris, France; Sorbonne Université, F-75005 Paris, France*; orcid.org/0000-0001-8584-6547

Pengyu Ren − *Department of Biomedical Engineering, The University of Texas at Austin, Austin, Texas 78712, United States*; orcid.org/0000-0002-5613-1910

Jay W. Ponder − *Department of Chemistry, Washington University in Saint Louis, Saint Louis, Missouri 63110, United States*; orcid.org/0000-0001-5450-9230

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.0c01164

Notes

The authors declare no competing financial interest.

Code Availability: The present code has been released in phase advance in link with the High Performance Computing community COVID-19 research efforts. The software is freely accessible to Academics via GitHub: https://github.com/TinkerTools/tinker-hp

ACKNOWLEDGMENTS

This work was made possible thanks to funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 810367), project EMC2. This project was initiated in 2019 with a "Contrat de Progrès" grant from GENCI (France) in collaboration with HPE and NVIDIA to port Tinker-HP on the Jean-Zay HPE SGI 8600 GPUs system (IDRIS supercomputer center, GENCI-CNRS, Orsay, France) using OpenACC. F.C. acknowledges funding from the French state funds managed by the CalSimLab LABEX and the ANR within the Investissements d'Avenir program (reference ANR11-IDEX-0004-02) and support from the Direction Générale de l'Armement (DGA) Maîtrise NRBC of the French Ministry of Defense. Computations have been performed at GENCI on the Jean Zay machine (IDRIS) on grant no. A0070707671 and on the Irène Joliot Curie ATOS Sequana X1000 supercomputer (TGCC, Bruyères le Chatel, CEA, France) thanks to PRACE COVID-19 special allocation (projet COVID-HP). We thank NVIDIA (Romuald Josien and François Courteille, NVIDIA France) for offering us access to A100 supercomputer systems (DGX-A100 and Selene DGX-A100 SuperPod machines). P.R. and J.W.P. are grateful for support by National Institutes of Health (R01GM106137 and R01GM114237).

REFERENCES

(1) Hollingsworth, S. A.; Dror, R. O. Molecular Dynamics Simulation for All. *Neuron* **2018**, *99*, 1129−1143.

(2) Dror, R. O.; Dirks, R. M.; Grossman, J.; Xu, H.; Shaw, D. E. Biomolecular Simulation: A Computational Microscope for Molecular Biology. *Annu. Rev. Biophys.* **2012**, *41*, 429−452.

(3) Ponder, J. W.; Case, D. A. *Advances in protein chemistry*; Elsevier, 2003; Vol. *66*, pp 27−85.

(4) Huang, J.; Rauscher, S.; Nawrocki, G.; Ran, T.; Feig, M.; de Groot, B. L.; Grubmüller, H.; MacKerell, A. D. CHARMM36m: an improved force field for folded and intrinsically disordered proteins. *Nat. Methods* **2017**, *14*, 71−73.

(5) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. *J. Chem. Theory Comput.* **2015**, *11*, 3696−3713.

(6) Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *J. Am. Chem. Soc.* **1996**, *118*, 11225−11236.

(7) Oostenbrink, C.; Villa, A.; Mark, A. E.; Van Gunsteren, W. F. A biomolecular force field based on the free enthalpy of hydration and solvation: The GROMOS force-field parameter sets 53A5 and 53A6. *J. Comput. Chem.* **2004**, *25*, 1656−1676.

(8) Shi, Y.; Ren, P.; Schnieders, M.; Piquemal, J.-P. Polarizable Force Fields for Biomolecular Modeling. In *Reviews in Computational Chemistry Vol. 28*; John Wiley and Sons, Ltd., 2015; Chapter 2, pp 51−86. DOI: 10.1002/9781118889886.ch2.

(9) Jing, Z.; Liu, C.; Cheng, S. Y.; Qi, R.; Walker, B. D.; Piquemal, J.-P.; Ren, P. Polarizable Force Fields for Biomolecular Simulations: Recent Advances and Applications. *Annu. Rev. Biophys.* **2019**, *48*, 371−394.

(10) Melcr, J.; Piquemal, J.-P. Accurate biomolecular simulations account for electronic polarization. *Front. Mol. Biosci.* **2019**, *6*, 143.

(11) Bedrov, D.; Piquemal, J.-P.; Borodin, O.; MacKerell, A. D.; Roux, B.; Schröder, C. Molecular Dynamics Simulations of Ionic Liquids and Electrolytes Using Polarizable Force Fields. *Chem. Rev.* **2019**, *119*, 7940−7995.

(12) Lopes, P. E. M.; Huang, J.; Shim, J.; Luo, Y.; Li, H.; Roux, B.; MacKerell, A. D. Polarizable Force Field for Peptides and Proteins Based on the Classical Drude Oscillator. *J. Chem. Theory Comput.* **2013**, *9*, 5430−5449.

(13) Lemkul, J. A.; Huang, J.; Roux, B.; MacKerell, A. D. An Empirical Polarizable Force Field Based on the Classical Drude Oscillator Model: Development History and Recent Applications. *Chem. Rev.* **2016**, *116*, 4983−5013.

(14) Lin, F.-Y.; Huang, J.; Pandey, P.; Rupakheti, C.; Li, J.; Roux, B.; MacKerell, A. D. Further Optimization and Validation of the Classical Drude Polarizable Protein Force Field. *J. Chem. Theory Comput.* **2020**, *16*, 3221−3239.

(15) Ren, P. Y.; Ponder, J. W. Polarizable Atomic Multipole Water Model for Molecular Mechanics Simulation. *J. Phys. Chem. B* **2003**, *107*, 5933−5947.

(16) Shi, Y.; Xia, Z.; Zhang, J.; Best, R.; Wu, C.; Ponder, J. W.; Ren, P. Polarizable Atomic Multipole-Based AMOEBA Force Field for Proteins. *J. Chem. Theory Comput.* **2013**, *9*, 4046−4063.

(17) Zhang, C.; Lu, C.; Jing, Z.; Wu, C.; Piquemal, J.-P.; Ponder, J. W.; Ren, P. AMOEBA Polarizable Atomic Multipole Force Field for Nucleic Acids. *J. Chem. Theory Comput.* **2018**, *14*, 2084−2108.

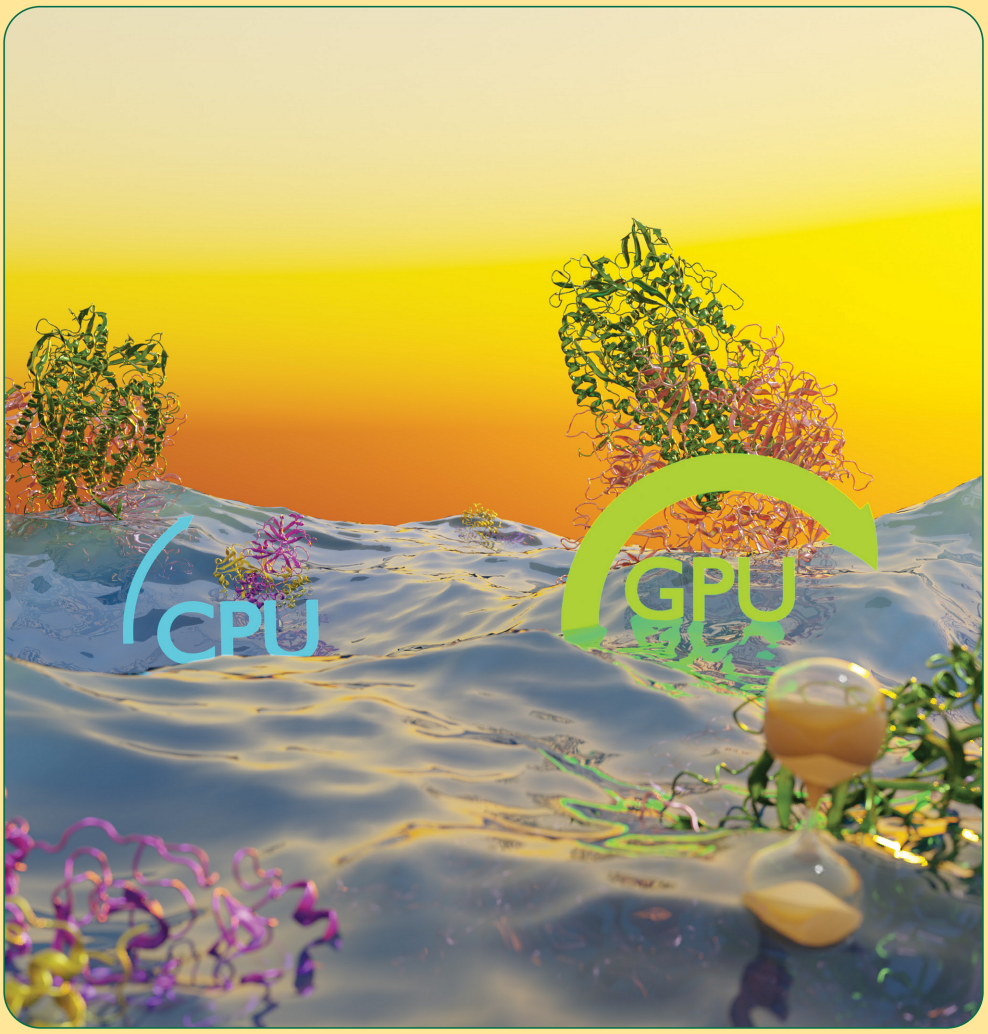
(18) Jiang, W.; Hardy, D. J.; Phillips, J. C.; MacKerell, A. D.; Schulten, K.; Roux, B. High-Performance Scalable Molecular Dynamics Simulations of a Polarizable Force Field Based on Classical Drude Oscillators in NAMD. *J. Phys. Chem. Lett.* **2011**, *2*, 87−92.

(19) Lemkul, J. A.; Roux, B.; van der Spoel, D.; MacKerell, A. D., Jr. Implementation of extended Lagrangian dynamics in GROMACS for polarizable simulations using the classical Drude oscillator model. *J. Comput. Chem.* **2015**, *36*, 1473−1479.
(20) Lagardère, L.; Jolly, L.-H.; Lipparini, F.; Aviat, F.; Stamm, B.; Jing, Z. F.; Harger, M.; Torabifard, H.; Cisneros, G. A.; Schnieders, M. J.; Gresh, N.; Maday, Y.; Ren, P. Y.; Ponder, J. W.; Piquemal, J.-P. Tinker-HP: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields. *Chem. Sci.* **2018**, *9*, 956−972.
(21) Rackers, J. A.; Wang, Z.; Lu, C.; Laury, M. L.; Lagardère, L.; Schnieders, M. J.; Piquemal, J.-P.; Ren, P.; Ponder, J. W. Tinker 8: Software Tools for Molecular Design. *J. Chem. Theory Comput.* **2018**, *14*, 5273−5289.
(22) Jolly, L.-H.; Duran, A.; Lagardère, L.; Ponder, J. W.; Ren, P.; Piquemal, J.-P. Raising the Performance of the Tinker-HP Molecular Modeling Package [Article v1.0]. *LiveCoMS* **2019**, *1*, 10409.
(23) Stone, J. E.; Hardy, D. J.; Ufimtsev, I. S.; Schulten, K. GPU-accelerated molecular modeling coming of age. *J. Mol. Graphics Modell.* **2010**, *29*, 116−125.
(24) Götz, A. W.; Williamson, M. J.; Xu, D.; Poole, D.; Le Grand, S.; Walker, R. C. Routine Microsecond Molecular Dynamics Simulations with AMBER on GPUs. 1. Generalized Born. *J. Chem. Theory Comput.* **2012**, *8*, 1542−1555.
(25) Páll, S.; Zhmurov, A.; Bauer, P.; Abraham, M.; Lundborg, M.; Gray, A.; Hess, B.; Lindahl, E. Heterogeneous Parallelization and Acceleration of Molecular Dynamics Simulations in GROMACS. *J. Chem. Phys.* **2020**, *153*, 134110.
(26) Salomon-Ferrer, R.; Götz, A. W.; Poole, D.; Le Grand, S.; Walker, R. C. Routine Microsecond Molecular Dynamics Simulations with AMBER on GPUs. 2. Explicit Solvent Particle Mesh Ewald. *J. Chem. Theory Comput.* **2013**, *9*, 3878−3888.
(27) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L.-P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Comput. Biol.* **2017**, *13*, e1005659.
(28) Harger, M.; Li, D.; Wang, Z.; Dalby, K.; Lagardère, L.; Piquemal, J.-P.; Ponder, J.; Ren, P. Tinker-OpenMM: Absolute and relative alchemical free energies using AMOEBA on GPUs. *J. Comput. Chem.* **2017**, *38*, 2047−2055.
(29) Potluri, S.; Luehr, N.; Sakharnykh, N. Simplifying Multi-GPU Communication with NVSHMEM. *GPU Technology Conference*; NVIDIA, 2016.
(30) Frenkel, D.; Smit, B. *Understanding molecular simulation: from algorithms to applications*; Elsevier, 2001; Vol. *1*.
(31) Wienke, S.; Springer, P.; Terboven, C.; an Mey, D. OpenACC—First Experiences with Real-World Applications. In *Euro-Par 2012 Parallel Processing*. Euro-Par 2012. Lecture Notes in Computer Science; Kaklamanis, C., Papatheodorou, T., Spirakis, P. G., Eds.; Springer: Berlin, Heidelberg, 2012; Vol. *7484*, pp 859−870 DOI: 10.1007/978-3-642-32820-6_85.
(32) Chandrasekaran, S.; Juckeland, G. *OpenACC for Programmers: Concepts and Strategies*, 1st ed.; Addison-Wesley Professional, 2017.
(33) Sanders, J.; Kandrot, E. *CUDA by example: an introduction to general-purpose GPU programming*; Addison-Wesley Professional, 2010.
(34) Volkov, V. Understanding Latency Hiding on GPUs. *Ph.D. thesis*, EECS Department, University of California, Berkeley, 2016.
(35) Kraus, J. An introduction to CUDA-aware MPI. *NVIDIA Developer Blog*; 2013; https://developer.nvidia.com/blog/introduction-cuda-aware-mpi/.
(36) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A smooth particle mesh Ewald method. *J. Chem. Phys.* **1995**, *103*, 8577−8593.
(37) Lagardère, L.; Lipparini, F.; Polack, E.; Stamm, B.; Cancès, E.; Schnieders, M.; Ren, P.; Maday, Y.; Piquemal, J.-P. Scalable Evaluation of Polarization Energy and Associated Forces in Polarizable Molecular Dynamics: II. Toward Massively Parallel Computations Using Smooth Particle Mesh Ewald. *J. Chem. Theory Comput.* **2015**, *11*, 2589−2599.
(38) NVIDIA Corporation. *CUDA Toolkit 11.1 CUFFT Library Programming Guide 2020*; NVIDIA, 2020;http://developer.nvidia.com/nvidia-gpu-computing-documentation.
(39) Lipparini, F.; Lagardère, L.; Stamm, B.; Cancès, E.; Schnieders, M.; Ren, P.; Maday, Y.; Piquemal, J.-P. Scalable Evaluation of Polarization Energy and Associated Forces in Polarizable Molecular Dynamics: I. Toward Massively Parallel Direct Space Computations. *J. Chem. Theory Comput.* **2014**, *10*, 1638−1651.
(40) Phillips, J. C.; Zheng, Gengbin; Kumar, S.; Kale, L. V. NAMD: Biomolecular Simulation on Thousands of Processors. *SC '02: Proceedings of the 2002 ACM/IEEE Conference on Supercomputing*; ACM, 2002; pp 36−36.
(41) Tuckerman, M.; Berne, B. J.; Martyna, G. J. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.* **1992**, *97*, 1990−2001.
(42) Bussi, G.; Donadio, D.; Parrinello, M. Canonical sampling through velocity rescaling. *J. Chem. Phys.* **2007**, *126*, 014101.
(43) Zhou, J.; Ross, K. A. Implementing database operations using SIMD instructions. *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*; ACM, 2002; pp 145−156.
(44) Nickolls, J.; Dally, W. J. The GPU computing era. *IEEE micro* **2010**, *30*, 56−69.
(45) Le Grand, S.; Götz, A. W.; Walker, R. C. SPFP: Speed without compromise—A mixed precision model for GPU accelerated molecular dynamics simulations. *Comput. Phys. Commun.* **2013**, *184*, 374−380.
(46) Yates, R. Fixed-point arithmetic: An introduction. *Digital Signal Labs* **2009**, *81*, 198.
(47) Bowers, K. J.; Dror, R. O.; Shaw, D. E. The midpoint method for parallelization of particle simulations. *J. Chem. Phys.* **2006**, *124*, 184109.
(48) Lagardère, L.; Aviat, F.; Piquemal, J.-P. Pushing the Limits of Multiple-Time-Step Strategies for Polarizable Point Dipole Molecular Dynamics. *J. Phys. Chem. Lett.* **2019**, *10*, 2593−2599.
(49) Célerse, F.; Lagardère, L.; Derat, E.; Piquemal, J.-P. Massively parallel implementation of Steered Molecular Dynamics in Tinker-HP:comparisons of polarizable and non-polarizable simulations of realistic systems. *J. Chem. Theory Comput.* **2019**, *15*, 3694−3709.
(50) Miao, Y.; Feher, V. A.; McCammon, J. A. Gaussian accelerated molecular dynamics: Unconstrained enhanced sampling and free energy calculation. *J. Chem. Theory Comput.* **2015**, *11*, 3584−3595.
(51) Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *J. Am. Chem. Soc.* **1996**, *117*, 11225−11236.
(52) Jaffrelot-Inizan, T.; Célerse, F.; Adjoua, O.; El Ahdab, D.; Jolly, L.-H.; Liu, C.; Ren, P.; Montes, M.; Lagarde, N.; Lagardère, L. High-Resolution Mining of SARS-CoV-2 Main Protease Conformational Space: Supercomputer-Driven Unsupervised Adaptive Sampling. *Chem. Sci.* **2021**, DOI: 10.1039/D1SC00145K.
(53) Chapman, B.; Curtis, T.; Pophale, S.; Poole, S.; Kuehn, J.; Koelbel, C.; Smith, L. Introducing OpenSHMEM: SHMEM for the PGAS community. *Proceedings of the Fourth Conference on Partitioned Global Address Space Programming Model*; ACM, 2010; pp 1−3.
(54) Liu, C.; Piquemal, J.-P.; Ren, P. AMOEBA+ Classical Potential for Modeling Molecular Interactions. *J. Chem. Theory Comput.* **2019**, *15*, 4122−4139.
(55) Liu, C.; Piquemal, J.-P.; Ren, P. Implementation of Geometry-Dependent Charge Flux into the Polarizable AMOEBA+ Potential. *J. Phys. Chem. Lett.* **2020**, *11*, 419−426.
(56) Gresh, N.; Cisneros, G. A.; Darden, T. A.; Piquemal, J.-P. Anisotropic, polarizable molecular mechanics studies of inter-, intramolecular interactions, and ligand-macromolecule complexes. A bottom-up strategy. *J. Chem. Theory Comput.* **2007**, *3*, 1960−1986.
(57) Bonomi, M.; Bussi, G.; Camilloni, C.; Tribello, G. A.; Banáš, P.; Barducci, A.; Bernetti, M.; Bolhuis, P. G.; Bottaro, S.; Branduardi, D.; et al. Promoting transparency and reproducibility in enhanced molecular simulations. *Nat. Methods* **2019**, *16*, 670−673.

Conclusion

The Tinker-HP platform, which utilizes multiple GPUs and multi-precision techniques, has proven to be highly accurate and scalable. This has resulted in an improved performance and new capabilities for dealing with long time scale simulations on large, realistic systems using AMOEBA. The approach greatly reduces the time required for routine simulations, which would have previously required thousands of CPUs on a single GPU card. Additionally, the GPU-accelerated Tinker-HP provides the best performance ever obtained for AMOEBA simulations, extending the applicability of PFFs to μs long simulations. Thanks to GPUs, computations that would have taken years can now be achieved in just a few days. Finally, this work paves the way for coupling FFs with machine learning potential models and nuclear quantum effects.

JCTC
Journal of Chemical Theory and Computation
April 2021 Volume 17 Number 4 pubs.acs.org/JCTC



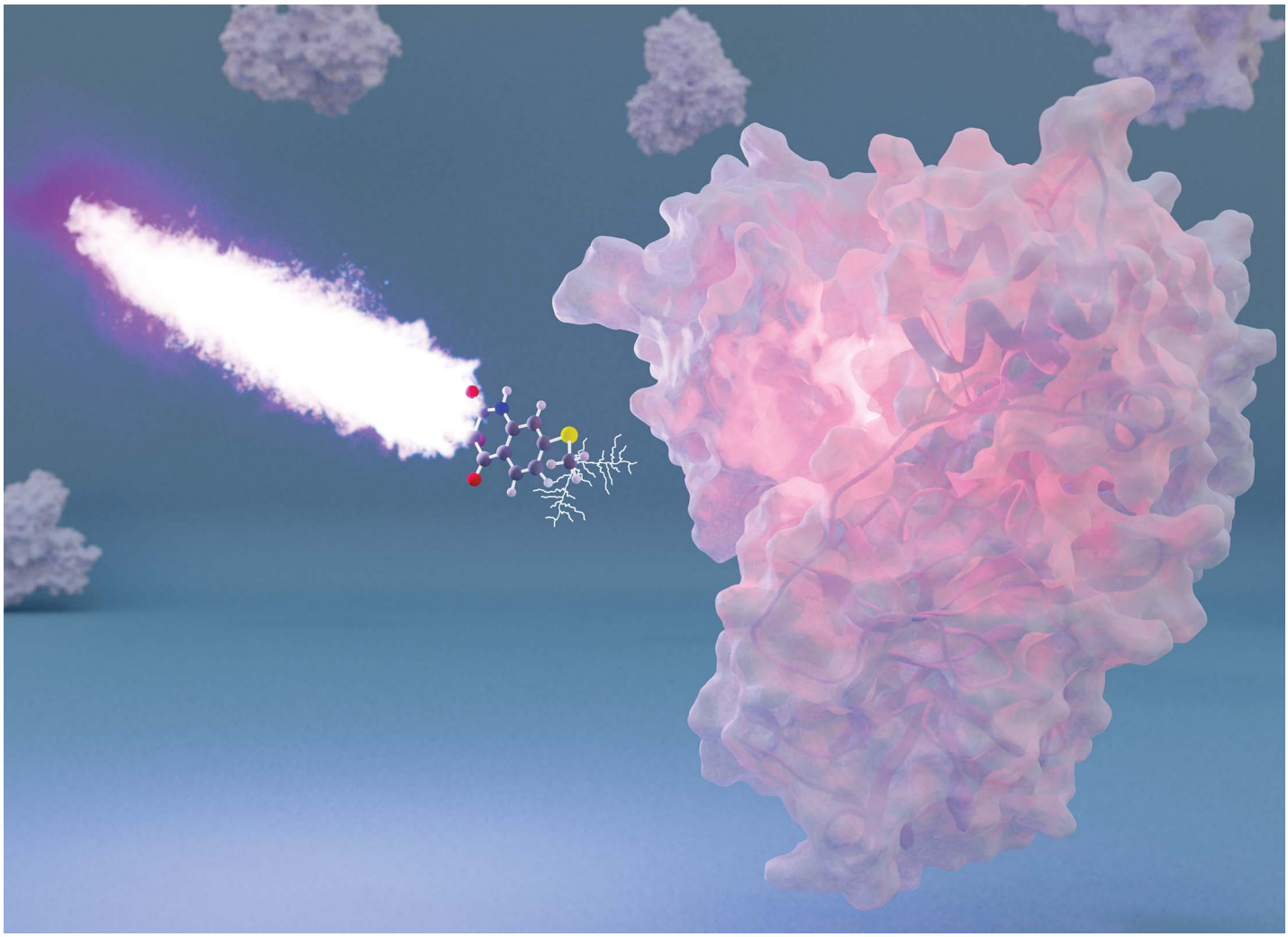
ACS Publications
Most Trusted. Most Cited. Most Read.
www.acs.org

B

Advancing the Discovery of SARS-CoV-2 M^{pro} Inhibitors through Computationally Driven Approaches and Deep Learning-Driven Markov State Models

Introduction

This article [174] provide a computational approach for discovering and analyzing binding modes of inhibitors targeting the SARS-CoV-2 M^{pro} enzyme, using simulations data from Chapter 3 Section 3.1. By conducting binding free energy calculations and extensive unsupervised adaptive sampling simulation on the ligand-binding site, it closely examined specific subpockets of the M^{pro}'s substrate binding site. The insights gained from this analysis were then used to design and synthesize both non-covalent and covalent inhibitors. In the context of this thesis, we analyzed the binding conformations of the best compound by employing a combination of the k-means clustering method and deep learning-driven Hidden Markov State Models. Specifically, we utilized the time-lagged variational autoencoder, which was developed by Noé et al., to enhance the accuracy of our analysis.[149, 150, 89]



Showcasing research from Professor Piquemal's laboratory, Department of Chemistry, Sorbonne Université, Paris, France and from Qubit Pharmaceuticals, Paris, France.

Computationally driven discovery of SARS-CoV-2 M^{pro} inhibitors: from design to experimental validation

We report a fast-track computationally driven discovery of new SARS-CoV-2 main protease (Mpro) inhibitors whose potency ranges from mM for the initial non-covalent ligands to sub-μM for the final covalent compound (IC50 = 830 ± 50 nM). The project extensively relied on high-resolution all-atom molecular dynamics simulations and absolute binding free energy calculations performed using the polarizable AMOEBA force field. While simulations extensively use high performance computing to strongly reduce the time-to-solution, they were systematically coupled to nuclear magnetic resonance experiments to drive synthesis and for in vitro characterization of compounds.

As featured in:

See Jean-Philip Piquemal, Davide Sabbadin *et al.*, *Chem. Sci.*, 2022, **13**, 3674.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/chemical-science

Registered charity number: 207890

Chemical Science

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE

View Article Online
View Journal | View Issue

Cite this: *Chem. Sci.*, 2022, **13**, 3674

All publication charges for this article have been paid for by the Royal Society of Chemistry

Computationally driven discovery of SARS-CoV-2 M^{pro} inhibitors: from design to experimental validation†‡

Léa El Khoury, §[a] Zhifeng Jing, §[a] Alberto Cuzzolin,[b] Alessandro Deplano,[c] Daniele Loco,[a] Boris Sattarov,[a] Florent Hédin,[a] Sebastian Wendeborn,[d] Chris Ho,[a] Dina El Ahdab,[n] Theo Jaffrelot Inizan,[n] Mattia Sturlese,[g] Alice Sosic,[e] Martina Volpiana,[e] Angela Lugato,[e] Marco Barone,[e] Barbara Gatto,[e] Maria Ludovica Macchia,[e] Massimo Bellanda,[f] Roberto Battistutta,[f] Cristiano Salata,[h] Ivan Kondratov,[i] Rustam Iminov,[i] Andrii Khairulin,[i] Yaroslav Mykhalonok,[i] Anton Pochepko,[i] Volodymyr Chashka-Ratushnyi,[i] Iaroslava Kos,[i] Stefano Moro,[g] Matthieu Montes,[j] Pengyu Ren,[k] Jay W. Ponder,[lm] Louis Lagardère,[n] Jean-Philip Piquemal*[no] and Davide Sabbadin*[a]

We report a fast-track computationally driven discovery of new SARS-CoV-2 main protease (M^{pro}) inhibitors whose potency ranges from mM for the initial non-covalent ligands to sub-μM for the final covalent compound (IC_{50} = 830 ± 50 nM). The project extensively relied on high-resolution all-atom molecular dynamics simulations and absolute binding free energy calculations performed using the polarizable AMOEBA force field. The study is complemented by extensive adaptive sampling simulations that are used to rationalize the different ligand binding poses through the explicit reconstruction of the ligand–protein conformation space. Machine learning predictions are also performed to predict selected compound properties. While simulations extensively use high performance computing to strongly reduce the time-to-solution, they were systematically coupled to nuclear magnetic resonance experiments to drive synthesis and for *in vitro* characterization of compounds. Such a study highlights the power of *in silico* strategies that rely on structure-based approaches for drug design and allows the protein conformational multiplicity problem to be addressed. The proposed fluorinated tetrahydroquinolines open routes for further optimization of M^{pro} inhibitors towards low nM affinities.

Received 25th October 2021
Accepted 3rd February 2022

DOI: 10.1039/d1sc05892d

rsc.li/chemical-science

1. Introduction

Since December 2019, the COVID-19 global pandemic has put the entire world on edge.[1,2] The disease is due to a coronavirus (CoV) called SARS-CoV-2 (severe acute respiratory syndrome, SARS) that has triggered the start of an unprecedented research effort.[3–5] While the vaccination strategy[6] has been particularly successful with the rise of mRNA techniques, additional programs have

[a]*Qubit Pharmaceuticals, Incubateur Paris Biotech Santé, 24 Rue du Faubourg Saint Jacques, 75014 Paris, France. E-mail: davide@qubit-pharmaceuticals.com*
[b]*Chiesi Farmaceutici S.p.A, Nuovo Centro Ricerche, Largo Belloli 11a, 43122, Parma, Italy*
[c]*Pharmacelera, Torre R, 4a planta, Despatx A05, Parc Cientific de Barcelona, Baldiri Reixac 8, 08028 Barcelona, Spain*
[d]*University of Applied Sciences and Arts Northwestern Switzerland, School of LifeSciences, Hofackerstrasse 30, CH-4132 Muttenz, Switzerland*
[e]*Department of Pharmaceutical and Pharmacological Sciences, University of Padova, via Marzolo 5, 35131, Padova, Italy*
[f]*Department of Chemistry, University of Padova, via Marzolo 1, 35131, Padova, Italy*
[g]*Molecular Modeling Section, Department of Pharmaceutical and Pharmacological Sciences, University of Padua, via F. Marzolo 5, 35131, Padova, Italy*
[h]*Department of Molecular Medicine, University of Padua, via Gabelli 63, 35121, Padova, Italy*
[i]*Enamine Ltd, 78 Chervonotkats'ka Str., Kyiv 02094, Ukraine*
[j]*Laboratoire GBCM, EA7528, Conservatoire National des Arts et Métiers, Hesam Université, 2 Rue Conte, 75003 Paris, France*
[k]*University of Texas at Austin, Department of Biomedical Engineering, TX 78712, USA*
[l]*Department of Chemistry, Washington University in Saint Louis, MO 63130, USA*
[m]*Department of Biochemistry and Molecular Biophysics, Washington University School of Medicine, MO 63110, USA*
[n]*Sorbonne Université, Laboratoire de Chimie Théorique, UMR 7616 CNRS, 75005, Paris, France. E-mail: jean-philip.piquemal@sorbonne-universite.fr*
[o]*Institut Universitaire de France, 75005, Paris, France*

† Qubit Pharmaceuticals and Sorbonne Université have submitted a preliminary patent application on the compounds reported in this study.

‡ Electronic supplementary information (ESI) available. See DOI: 10.1039/d1sc05892d

§ These authors contributed equally to this work.

been launched to obtain antivirals able to reduce the impact of COVID-19 on ill patients. Despite these efforts, few potential treatments are presently available with the exception of Paxlovid, a nirmatrelvir/ritonavir combo proposed by Pfizer.[7] Due to the persistence of the pandemic, it remains essential to propose new antiviral drugs. A possible strategy consists in designing small molecules to interact with one of the main proteins of the SARS-CoV-2 virus, thus blocking its activity. Among the potential targets, the main protease protein, denoted as M^{pro} or $3CL^{pro}$, is a primary choice[8] as it has no human homolog and it is well conserved among coronaviruses,[9] especially in terms of the structure of its active site, catalytic dyad, and dimer interface. Furthermore, M^{pro} is required to release viral proteins for particle assembly, and is thus essential to the virus replication cycle.

Developing a new drug targeting the viral M^{pro} is challenging as it requires extensive resources and the success rate is notoriously low.[10] Relying on *in silico* driven rational design could accelerate the process. In fact, it diminishes the cost by reducing the need for synthetic iterations while also providing an interpretation of the interactions occurring between the target protein and potential inhibitors.

It is important to note that theoretical modeling of M^{pro} is challenging as the protein exhibits high structural flexibility[11–13] leading to high conformational complexity. M^{pro} is also involved in a variety of complex protein–ligand–solvent interaction networks.[12,13] These challenges can be tackled using a high-resolution modeling approach[12,13] going beyond rigid docking procedures (see ref. 14 for a detailed discussion of the difficulties of docking approaches in predicting the native binding modes of small molecules within M^{pro}).

Many studies have been devoted to the design of new M^{pro} inhibitors[3,5,15–25] through joint computational and experimental approaches. In particular, a recent study by the Jorgensen group highlighted the usefulness of relative binding free energy (RBFE) computations as part of the drug design process.[26]

In this paper, we present a computationally driven discovery and binding mode rationalization of new SARS-CoV-2 M^{pro} inhibitors. In doing so, we build on our previous high-resolution M^{pro} molecular dynamics studies.[12,13] Here, we explore more deeply some specific subpockets of the substrate binding site of the protease using absolute binding free energy (ABFE) calculations and adaptive sampling grounded on extensive molecular dynamics simulations with high-resolution polarizable force fields (PFFs). Using the GPU-accelerated module[27] (GPU = Graphics Processing Unit) of the Tinker-HP molecular dynamics package[28] coupled to the AMOEBA PFF,[29–32] it has been shown that simulations can reach the required level of accuracy and μs timescales needed to explore the structural rearrangement and interactions profile of this flexible protein.[12,13] More precisely, the modeling of M^{pro} necessitates the ability to evaluate at high resolution various types of key interactions including hydrogen bonds, salt bridges, π–π stacking, and specific solvation effects. Long timescales are required to achieve sufficient sampling. This is now possible by using the large number of graphics processing units (GPUs) that are presently available on supercomputers and high-performance cloud computing platforms. In this study, we combine our computationally driven strategy, using absolute binding free energy computations[33–37] and unsupervised adaptive sampling,[12,13] with machine learning-assisted property predictions, while conducting extensive characterization experiments including nuclear magnetic resonance (NMR), mass spectrometry (MS), and FRET-based assays to evaluate the activity of the newly designed compounds.

In the following, we introduce our design strategy, which led to non-covalent and covalent inhibitors of M^{pro} (ESI-Fig. 1‡). Then, we describe how an interplay between experiments and molecular simulations allowed the discovery of a final compound (QUB-00006-Int-07) with a high affinity to the protease ($IC_{50} = 830 \pm 50$ nM).

2. Computational details

2.1. Systems preparation

The protease dimer structure (PDB code: 7L11) was used for all the MD simulations and it was prepared at physiological pH (pH = 7). This structure has a higher resolution (1.80 Å) than the PDB structure (PDB code: 6LU7) used in our previous work[12] (resolution of 2.16 Å). Both structures are of the holo state in complex with covalent inhibitors, and the rotamers of the key residues at the catalytic site (Cys145, His41, His162, His163, and His172) are virtually identical. The protonation states of His residues were assigned based on previous work,[38] where His41 and His80 are protonated at the delta carbon atom and all other His residues are epsilon-protonated, which is favorable for substrate binding.[38] This is different from our previous work where His64 and His80 are protonated at the delta carbon atom and all other histidines are epsilon-protonated.[12] All water molecules were retained except for those that might collide with the ligands.

2.2. Simulation protocols

All-atom simulations were performed using Qubit Pharmaceuticals' Atlas platform which enables the use of any type of High-Performance Computing (HPC) system including cloud supercomputing infrastructures. Among its possibilities, Atlas has the ability to efficiently handle polarizable force field molecular dynamics simulations using a custom version of the multi-GPU module[27] of the Tinker-HP molecular dynamics package,[28,39] to perform docking runs using either Autodock-Vina[40] or Autodock-GPU,[41] and to enable machine learning predictions of molecular properties.

2.2.1. Molecular dynamics simulations. All Tinker-HP MD simulations (for a total of several μs) were performed in mixed precision to benefit from a strong acceleration of simulations using GPUs.[27] The AMOEBA polarizable force field[29–32] was used to describe the full systems including the protein, ions and water. Several utilities (TinkerTools) from Tinker 8 (ref. 42) were used. Periodic boundary conditions were applied within the framework of smooth particle mesh Ewald summation[43,44] with a grid of dimensions 120 × 120 × 120 using a cubic box with side lengths of 97 Å. The Ewald cutoff was set to 7 Å, and the van der Waals cutoff was 12 Å. Langevin molecular dynamics simulations were performed using the recently introduced BAOAB-RESPA1 integrator (10 fs outer timestep),[45]

a preconditioned conjugate gradient polarization solver (with a 10^{-5} convergence threshold) to solve polarization at each time step,[46] hydrogen-mass repartitioning (HMR) and random initial velocities. Absolute binding free energy simulations following a protocol described in the next section were performed as well as adaptive sampling runs that are also described further in the text. Absolute binding free energy computations were both performed on the HPE Jean Zay Supercomputer (IDRIS, GENCI, France) and on Amazon Web Services (AWS). All adaptive sampling computations were performed using AWS. Simulations on AWS used both p3.2x (NVIDIA V100 GPU cards) and p4d.24xlarge (NVIDIA A100 GPU cards) instances whereas computations on the Jean Zay supercomputer were powered by V100 cards.

2.2.2. Molecular docking protocol. The protonation states of the ligands were calculated at a neutral pH and the hydrogen atoms were added using Chimera. Next, we docked the ligands QUB-00006-Int-01(*R*) and QUB-00006-Int-01(*S*) into the M^{pro} dimer structure using Autodock Vina 1.1.2.[40] AutoDock Vina requires the pdbqt format for the input files of the receptor and the ligand. Therefore, using the scripts 'prepare_receptor4.py' (v 1.13) and 'prepare_ligand4.py' (v 1.10) provided by Autodock Tools,[47] we generated pdbqt files corresponding to the receptor and the ligands, respectively. We set the exhaustiveness search to 100 and the num_mode option to 50.

Since molecular docking could suggest reasonable potential binding modes, but does not always rank the most likely binding mode as the best docked pose,[14,48] we visually inspected the generated docked poses and chose an ensemble of binding poses with different binding orientations that we used to run MD and ABFE calculations in order to explore the binding mode of QUB-00006-Int-01, as described in the Results and discussion section.

2.2.3. Equilibration. A detailed description of the equilibration protocol used for MD simulations can be found in the ESI.‡

2.2.4. High-resolution adaptive sampling simulations. Starting from several binding poses as described above we ran adaptive sampling simulations using the AMOEBA force field[29–32] in order to explore their stability and more generally to explore the conformational space of the ligands in the pocket of the M^{pro}. Because of the flexibility of the pocket and the role it may play in the exploration of the potential binding modes of the ligand, we chose to keep the whole system (ligand + protein) flexible during this sampling phase. The restart strategy (similar to the one introduced in ref. 12) was the following: first, all the previously generated conformations of the protein were loaded and aligned with MDTraj,[49] then PCAs of the conformations of the ligand were computed using Scikitlearn[50] and these frames were projected on the first four PCAs. Finally, the same scheme as the one described in ref. 12 was used to generate new starting points, favoring points that were less explored during the previous phases. In practice, a first set of 5 simulations of 10 nanoseconds were performed using different random seeds, and then 4 iterations of 10 times 10 nanoseconds were generated using the adaptive sampling protocol described above, for a total of 450 nanoseconds.

2.2.5. Absolute binding free energy calculations. In order to benefit from the high-accuracy evaluation of free energies using the AMOEBA force field,[33–37] we used the same clustering algorithms as described above to analyze the adaptive molecular dynamics simulations. The largest clusters were used for absolute free energy calculations. The double-decoupling protocol and the Bennett acceptance ratio (BAR)[51] method were used to calculate the standard binding free energy for each binding pose.[33,37] There were 27 or 26 thermodynamic states for the decoupling in the complex phase or the aqueous phase. A distance restraint between two groups of atoms in the ligand and in the protein binding pocket was applied when decoupling the ligand in the complex to accelerate the convergence when the ligand is fully decoupled, and the restraint was removed at an additional step at the full interaction state. A harmonic restraint with a force constant of 15.0 kcal mol^{-1} $Å^{-2}$ and radius of 2.0 Å was used. An analytical correction was added to the binding free energy to account for the standard state at 1.0 mol L^{-1} in the fully decoupled state. 10 ns simulations were performed for each thermodynamic state for the simulations of M^{pro} in complex with x0195, QUB-00006(*S*), QUB-00006(*R*), and QUB-00006-Int-07. For the simulations of M^{pro} in complex with QUB-00006-Int-01(*R*) and QUB-00006-Int-01(*S*), we ran each thermodynamic state for 20 ns. We used the BAOAB-RESPA1 integrator with a 10 fs time step and we calculated the electrostatic interactions using Ewald summation with a real space cutoff of 7 Å. van der Waals interactions were calculated using a cutoff of 12 Å with long-range correction.

2.3. Quantitative structure–property relationship (QSPR) modeling: predicting solubility using machine learning

Qubit Pharmaceuticals' Atlas internal machine learning-based QSPR module was used to predict the water solubility (log *S*, *S* measured in mol L^{-1}) and octanol/water partition coefficient (log *P*). To build a water solubility QSPR predictor, the AqSolDB dataset[52] was used as a training set. To predict octanol/water partition coefficients (log *P*), the dataset from EPA's OPERA[53] was used as a training set.

Selected datasets were preprocessed and standardized to some extent by authors of the corresponding publications. However, the need for additional processing was identified when doing exploratory data analysis. We discarded compounds with less than two carbon atoms and kept molecules with molecular weight between 50 and 750 daltons. Additional rules of fragment standardization developed at Qubit Pharmaceuticals were applied.

2.3.1. Similarity analysis. Tanimoto similarity[54] to the x0195 compound was calculated for each molecule using the MAACS fingerprint from the RDKit Open-Source Cheminformatics Software (https://www.rdkit.org). The Morgan circular fingerprint[55] with radius = 2 and nBits = 2048 from RDKit was also tested and the results (not shown) exhibit a similar ranking of the compounds.

3. Experimental protocol

3.1. Recombinant expression of SARS-CoV-2 M^{pro} in *E. coli*

The plasmid pGEX-6P-1 encoding SARS-CoV-2 M^{pro}[56] was a generous gift from Prof. Rolf Hilgenfeld, University of Lübeck,

Lübeck, Germany. Protein expression and purification were adapted from Zhang *et al.*[56] The expression plasmid was transformed into *E. coli* strain BL21 (DE3) and then pre-cultured in YT medium at 37 °C (100 μg mL^{-1} ampicillin) overnight. The pre-culture was used to inoculate fresh YT medium supplied with an antibiotic and the cells were grown at 37 °C to an OD600 of 0.6–0.8 before induction of overexpression with 0.5 mM isopropyl-D-thiogalactoside (IPTG). After 5 h at 37 °C, cells were harvested by centrifugation (5000*g*, 4 °C, 15 min) and frozen. The pellets were resuspended in buffer A (20 mM Tris, 150 mM NaCl, pH 7.8) supplemented with lysozyme, DNase I and PMSF for the lysis. The lysate was clarified by centrifugation at 12 000*g* at 4 °C for 1 h and loaded onto a HisTrap HP column (GE Healthcare) equilibrated with 98% buffer A/2% buffer B (20 mM Tris, 150 mM NaCl, 500 mM imidazole, pH 7.8). The column was washed with 95% buffer A/5% buffer B and then His-tagged M^{pro} was eluted with a linear gradient of imidazole ranging from 25 mM to 500 mM. Pooled fractions containing the target protein were subjected to buffer exchange with buffer A using a HiPrep 26/10 desalting column (GE Healthcare). Next, PreScission protease was added to remove the C-terminal His tag (20 μg of PreScission protease per mg of target protein) at 12 °C overnight. Protein solution was loaded onto a HisTrap HP column connected to a GSTrap FF column (GE Healthcare) equilibrated in buffer A to remove the GST-tagged PreScission protease, the His-tag, and the uncleaved protein. M^{pro} was finally purified with a Superdex 75 prep-grade 16/60 (GE Healthcare) SEC column equilibrated with buffer C (20 mM Tris, 150 mM NaCl, 1 mM EDTA, 1 mM DTT, pH 7.8). Fractions containing the target protein at high purity were pooled, concentrated at 25 mg mL^{-1} and flash-frozen in liquid nitrogen for storage in small aliquots at −80 °C.

3.2. Protein characterization and enzymatic activity

The molecular mass of the recombinant SARS-CoV-2 M^{pro} was determined by direct infusion electrospray ionization mass spectrometry (ESI-MS) on a Xevo G2-XS QTOF mass spectrometer (Waters). Samples were diluted in 50% acetonitrile with 0.1% formic acid to achieve a final 1 μM concentration of protein. The detected species displayed a mass of 33 796.64 Da, which matches very closely the value of 33 796.81 Da calculated from the theoretical full-length protein sequence (residues 1–306). To characterize the enzymatic activity of our recombinant M^{pro}, we adopted a FRET-based assay using the fluorogenic substrate 5-FAM-AVLQ′SGFRK(DABCYL)K (ProteoGenix) harbouring the cleavage site of SARS-CoV-2 M^{pro} (′ indicates the cleavage site). The fluorescence of the intact peptide is very low since the fluorophore 5-FAM and the quencher Dabcyl are in close proximity. When the substrate is cleaved by the protease, the fluorophore and the quencher are separated, increasing the fluorescence signal. Freshly unfrozen recombinant SARS-CoV-2 M^{pro} was used in our assays. The assay was performed by mixing 0.05 μM M^{pro} with different concentrations of substrate (1–128 μM) in the reaction buffer (20 mM Tris–HCl, 100 mM NaCl, 1 mM EDTA and 1 mM DTT, pH 7.3) in the final volume of 100 μL. Fluorescence intensity (Ex = 485 nm/Em = 535 nm) was monitored at 37 °C with a Victor3 microplate reader (PerkinElmer) for 50 min. A calibration curve was created by measuring multiple concentrations (from 0.001 to 5 μM) of free fluorescein in a final volume of 100 μL reaction buffer. Initial velocities were determined from the linear section of the curve, and the corresponding relative fluorescence units per unit of time (ΔRFU/s) were converted to the amount of the cleaved substrate per unit of time (μM s^{-1}) by fitting to the calibration curve of free fluorescein. Inner-filter effect corrections were applied for the kinetic measurements according to ref. 57. The catalytic efficiency k_{cat}/k_m resulted in 4819 ± 399 s^{-1} M^{-1}, in line with literature data.[56,58]

3.3. Nuclear magnetic resonance

All the NMR screening experiments were performed with a Bruker Neo 600 MHz spectrometer, equipped with a nitrogen cooled 5 mm Prodigy CryoProbe at 298 K. The ligand binding was monitored by WaterLOGSY (wLogsy)[59] and Saturation Transfer Difference (STD)[60] experiments in the presence and in the absence of the protein. Samples contained 10 μM M^{pro} and 100 μM to 2 mM ligand dissolved in 150 mM NaCl, 20 mM phosphate, 5% D_2O, and 4% DMSO-d6 (pH = 7.3). WaterLOGSY experiments were performed with a 180° inversion pulse applied to the water signal at 4.7 ppm using a Gaussian-shaped selective pulse of 5 ms. Each WaterLOGSY spectrum was acquired with 320 scans, a mixing time of 1.5 s and a relaxation delay of 4.5 s. STD experiments were performed with 256 scans. Selective saturation of the protein at 0.4 ppm frequency was carried out by a 2 s pulse train (60 Gaussian pulses of 50 ms separated by 1 ms intervals) included in the relaxation delay and a 30 ms spin-lock was used to reduce the broad background protein signal. The estimation of the KD was achieved by a STD titration according to a previously reported procedure and fitting the curves using OriginPro 2018 (OriginPro version 2018 developed by OriginLab Corporation, Northampton, MA, USA). The water suppression was achieved by the excitation sculpting pulse scheme.

3.4. Screening of potential M^{pro} inhibitors and hits validation

A FRET-based assay employed to test the enzymatic activity of the recombinant SARS-CoV-2 M^{pro} was used to evaluate the ability of the compounds to inhibit its activity *in vitro*. In fact, inhibition of M^{pro} by the tested compounds results in a decrease of the fluorescence signal compared to the M^{pro} activity in the absence of an inhibitor. A preliminary screening was first performed at a single compound concentration to rapidly identify the ability of the compounds to inhibit M^{pro} activity and to rank them according to their inhibitory activity. The protein was diluted in the reaction buffer (20 mM Tris–HCl, 100 mM NaCl, 1 mM EDTA and 1 mM DTT, pH 7.3) and pipetted into a 96-well plate to a final protein concentration of 0.02 μM in a final volume of 100 μL. Each compound at the final concentration of 100 μM was incubated with M^{pro} for 20 minutes at room temperature. After incubation, the peptide substrate (5 μM final) was added to initiate the reaction which

was monitored for 50 min at 37 °C. The final DMSO amount was 3.75%. Two controls were prepared for each experiment: the peptide substrate in the absence of M^{pro} (0% M^{pro} activity, hence minimal fluorescence intensity detected) and the reaction mixture in the absence of the compounds (100% M^{pro} activity, therefore maximal fluorescence intensity detected). Following the preliminary screening, the most active compounds (hits) were tested at increasing concentrations (0.25, 0.5, 1, 5, 25, 50, 100, and 150 μM) to determine the dose–response curves and calculate IC_{50} values fitted using GraphPad Prism 5 software. Each experiment was performed in triplicate and the results were used to calculate an average and a standard deviation.

3.5. Binding studies by mass spectrometry

Samples were prepared by mixing appropriate volumes of M^{pro} (10 μM final) with each compound in the reaction buffer (20 mM Tris–HCl, 100 mM NaCl, 1 mM EDTA and 1 mM DTT, pH 7.3). The final mixtures had a 1 : 1 or 10 : 1 compound : protein molar ratio. Samples were incubated at room temperature for 20 min before analysis. Control experiments were performed on 10 μM solutions of M^{pro} in the absence of the compounds. Mass spectrometric analyses were carried out in positive ion mode by ESI-MS under denaturing conditions, *i.e.* water/acetonitrile 50 : 50 with 0.1% formic acid on a Q-Tof Xevo G2S (Waters, Manchester, UK). Data were processed using MassLynx V4.1 software.

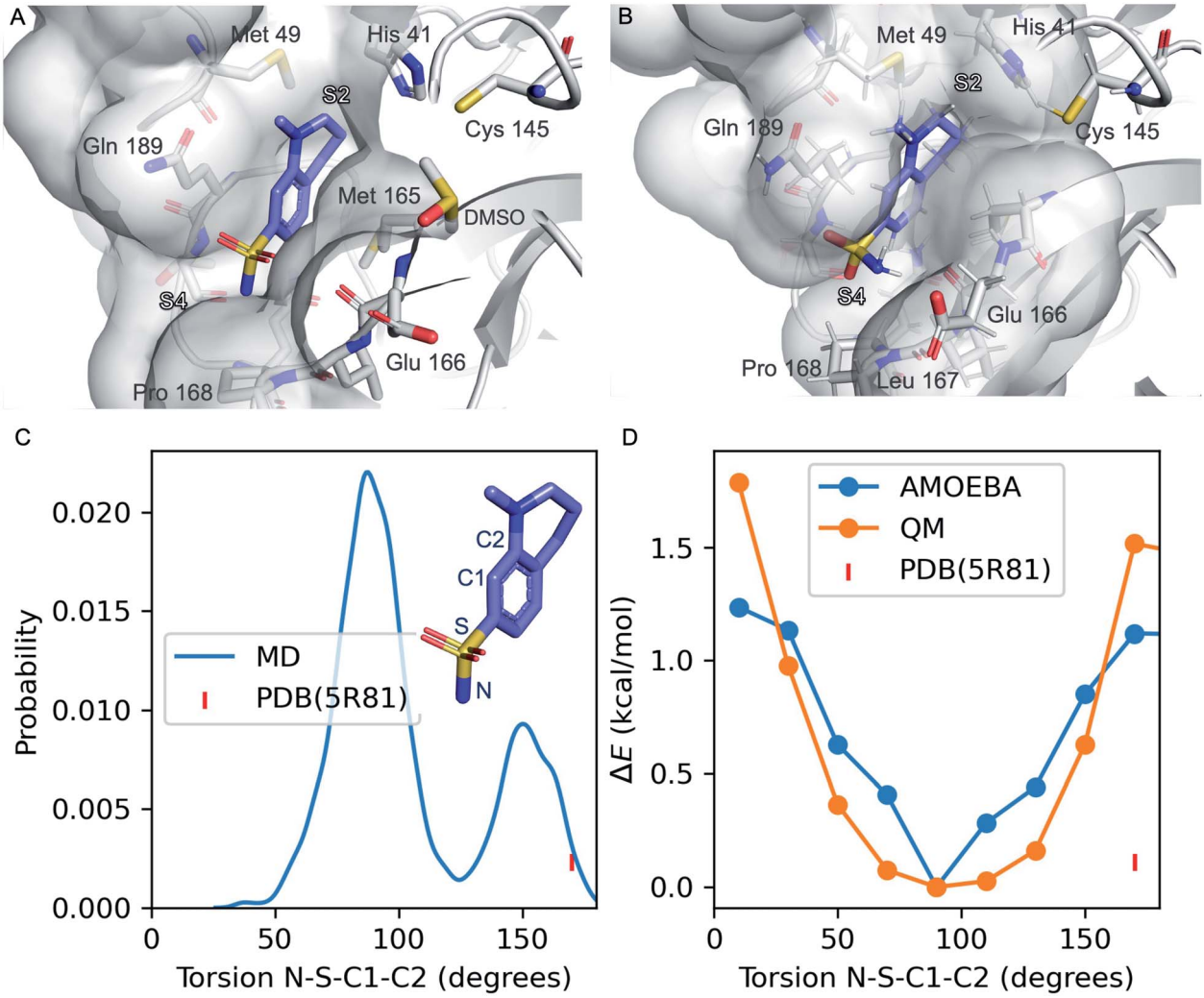


Fig. 1 Refinement of the co-crystal structure of x0195 and M^{pro} using MD simulations. (A) An unusual conformation of x0195 (in purple) located in the binding pocket formed by His41, Met49, Glu166, Gln189, and Pro168 and their surroundings (PDB code: 5R81), and (B) the relaxed structure of x0195 (in purple), obtained after the equilibration step, interacting with the amino acid residues of the substrate binding site. M^{pro} is shown in light grey. (C) Torsion angle distribution for the sulfonamide group during 20 ns of MD simulations (in blue) performed on the M^{pro} dimer in complex with x0195; the torsion angle of the sulfonamide group in the co-crystal structure is shown in pink. (D) Torsion energy scan calculated by AMOEBA (in blue) and QM (in orange); the torsion angle of the sulfonamide group in the co-crystal structure is shown in pink. QM level = ωB97x-D/6-31g*.[65–67]

3.6. Synthesis

The detailed synthetic protocol used to prepare all molecules can be found in the ESI.‡

4. Results and discussion

Several diverse fragments binding the viral M^{pro} have been identified by high-throughput crystallographic screening of this protease. Among the screened fragments, x0195 (PDB ID: 5R81 (ref. 61) – Fig. 1A) shows one of the highest binding affinities[62] and therefore provides a reasonable starting point for fragment-based design of novel M^{pro} inhibitors.

The crystal structure shows that x0195 is located within the M^{pro} substrate binding pocket, at the interface of the two subpockets S2 and S4 as described by Cannalire *et al.*[63] S4 is a solvent exposed subpocket that is partially composed of a flexible loop delimited by Gln189 and Gln192, while S2 is defined by the side chain residues of Phe140, Asn142, His163, Glu166, and His172, and the backbone atoms of Phe140 and Leu141.

In the co-crystal structure corresponding to M^{pro} in complex with x0195 (see Fig. 1A), the aromatic portion of the molecule is located between the side chains of Gln189 and Met 165, while the unsaturated region of the tetrahydroquinoline scaffold establishes a hydrophobic interaction with the side chains of His41 and Met49. The *N*-methyl group attached to the tetrahydroquinoline core is solvent exposed, while the sulfonamide moiety is in contact with Pro168 and Glu166. In particular, the aromatic ring of the small molecule is bisecting the SO_2 unit and the polar sulfonamide nitrogen ($-NH_2$) is reaching the boundaries of the hydrophobic part of the binding pocket composed of the alkyl chain of Pro168.

After comparing the available X-ray structural information with previously conducted studies on small molecule conformational preferences derived from crystal structure data,[64] we noticed that x0195 was modeled in a high energy conformation and that an unusual high-energy (*i.e.* repulsive) contact occurs between the sulfonamide oxygen and the carbonyl oxygen of the Glu166 backbone. Additionally, the tetrahydroquinoline scaffold was not fully exploring S2 subpocket boundaries. As reported by Cannalire *et al.*[63] and Zhang *et al.*,[8] the volume of the S2 subpocket in SARS-CoV M^{pro} is very similar to that of the MERS-CoV homologue. However, the volume of S2 in SARS-CoV M^{pro} (252 $Å^3$) is significantly larger than in other CoV homologues of the α-genus, such as the HCoV-NL63 M^{pro} (45 $Å^3$).[8,63] Therefore, exploiting this knowledge might be key to designing specific inhibitors of CoV M^{pro}.

In order to refine the available X-ray structural model and to gather more structural insights (*e.g.* protein flexibility and binding pocket rearrangements[12,13]) to guide the design of better binders of the subpocket S2, we ran all-atom molecular dynamics simulations using the AMOEBA polarizable force field[29–32] on M^{pro} (PDB code: 7L11) in complex with x0195 (PDB code: 5R81).

Our simulations show that the unusual high-energy contacts between the sulfonamide oxygen and the carbonyl oxygen of the Glu166 backbone no longer occurred. Also, regarding the electronic structure, we noticed that the p orbitals of the aromatic carbon C1 bisect (*e.g.* are parallel to) the SO_2 angle, compared with a 90° value for the same angle as reported in the crystal structure (see Fig. 1). Moreover, the NH_2 of the sulfonamide group is engaged in favorable polar interactions with the Gln189 side chain and the solvent.

Then, we performed absolute binding free energy calculations on the refined protein–ligand structure. Our results show that x0195 binds to the protein with a binding free energy of -2.83 kcal mol^{-1} at 283 K, which is comparable to the experimental binding energy (-3.59 ± 0.1 kcal mol^{-1}, see Table 1).

We obtained the experimental binding free energy by converting the experimental K_d (1.7 mM $\pm$ 0.2) provided in the literature[62] using the Gibbs free energy equation and the experimental temperature used in the binding assays (283 K). The agreement of the computed free energy prediction with the experimental results is reasonable. Further analysis of MD simulations suggests that the tetrahydroquinoline scaffold of x0195 is sub-optimally occupying the binding pocket.

We put in place design strategies to modify the chemical moieties of x0195 and potentially increase its binding affinity. Here, we introduce the design of a new molecule, namely QUB-00006 (Fig. 2), where we added two fluorines and a methyl group on the tetrahydroquinoline core of x0195. Also, we substituted the sulfonamide group on the aromatic ring of the molecule by a methanethiol. Fluorination at position 3 of the tetrahydroquinoline core could increase ligand occupancy with no disruption of the water network surrounding the binding pocket,[12,13] while methylation at position 4 seemed an interesting modification to increase the potential interactions of the ligand with binding pocket residues. QUB-00006 was generated based on the structure and position of x0195 in the co-crystal (5R81), and then placed in the receptor structure (M^{pro} dimer

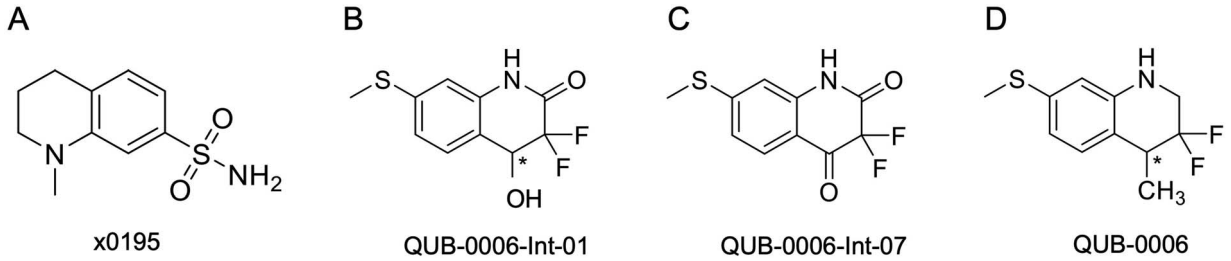


Fig. 2 2D structures of (A) x0195, (B) QUB-00006-Int-01, (C) QUB-00006-Int-07, and (D) QUB-00006. The asterisk represents a chiral center.

Table 1 Experimental and computed binding free energies (kcal mol^{-1}) for the non-covalent compounds. N.A. = not available (see text for details)

Compound	Computed ΔG	Experimental ΔG
QUB-00 006(*R*)	−2.73 ± 0.34	N.A.
QUB-00 006(*S*)	−2.72 ± 0.22	
QUB-00 006-Int-01(*R*)	−4.30 ± 0.35	−3.71 ± 0.2
QUB-00 006-Int-01(*S*)	−4.45 ± 0.29	
x0195	−2.83 ± 0.66	−3.59 ± 0.1
QUB-00 006-Int-07	−5.37 ± 0.23	Covalent binder

with the PDB code: 71LL); next, the M^{pro}–QUB-00006 complex was equilibrated using MD simulations (see ESI Section 1‡ for the detailed protocol), followed by free energy calculations. To explore the potential of our computational platform in designing new binders with no or few experimental data such as ligand–M^{pro} co-crystal structures, we leveraged all-atom molecular dynamics simulations on QUB-00006 complexed with M^{pro}. The aim of this approach is to gather insights on the binding conformation of the newly *in silico* designed ligand, assess pocket fitness, and evaluate its binding affinity using ABFE calculations.

The initial molecular conformation is mostly anchored at the binding pocket, with the α,α-difluoro-methyl group attached to the tetrahydroquinoline core fully occupying the buried part of the S2 subpocket, which is composed of the side chains of Met49 and His41, while the sulfonamide moiety extends to S4 (Leu167 and Pro168). We note that methylation at position 4 of the tetrahydroquinoline core introduces a chiral center, however no significant differences in terms of pocket occupancy between the *R* and *S* enantiomers were observed.

The computed absolute binding free energies for QUB-00006(*R*) and QUB-00006(*S*) are −2.73 ± 0.34 kcal mol^{-1} and −2.72 ± 0.22 kcal mol^{-1}, respectively (Table 1). These results suggest that the designed fluorinated fragment is a binder at the M^{pro} S2 subpocket and could represent a starting point for structure-based design of novel M^{pro} inhibitors.

The identified binding mode is defined by several favorable intermolecular interactions occurring between the newly designed ligand and the M^{pro} binding pocket: (i) the sulfur group of QUB-00006(*R*) interacts with the oxygen of the carbonyl belonging to the backbone of Glu166 with a distance of 3.3 Å, (ii) the α,α-difluoro moiety points towards His41, and (iii) the sulfur of Met49 establishes a favorable interaction with one of the two fluorines of the substrate (distance 3.3 Å). In fact, the sulfur–oxygen contact observed in our simulations is in agreement with the findings of a study conducted by Iwaoka *et al.*,[68] where they found that a total of 1200 and 626 fragments from the Cambridge Structural Database (CSD) and Protein Data Bank (PDB), respectively, have close intermolecular S–O contacts (with a distance of 3.52 Å or less). Another study analyzing the protein structures deposited in the Protein Data Bank reports 1133 interactions between His and halogen atoms found in 3833 PDB entries with one or more halogenated ligands co-crystallized with a protein.[69] Moreover, the strong S–F interaction identified during the simulations is in good agreement with experimentally observed distances for fluorine–sulfur contacts in crystal structures (2.8–3.4 Å).[70] It is worth noting that such interactions involving sulfur and halogen atoms are usually better captured with polarizable models than with their classical counterparts.[71–73]

QUB-00006 was then synthesized following the path in Fig. 4 in order to validate *in vitro* the simulation outcomes.

The ligand orientation in the MD simulations and the computed hydration ratio of the different atoms of QUB-00006 during ABFE simulations suggest that proton C is solvent exposed, while the protons of the methyl thioether group (group

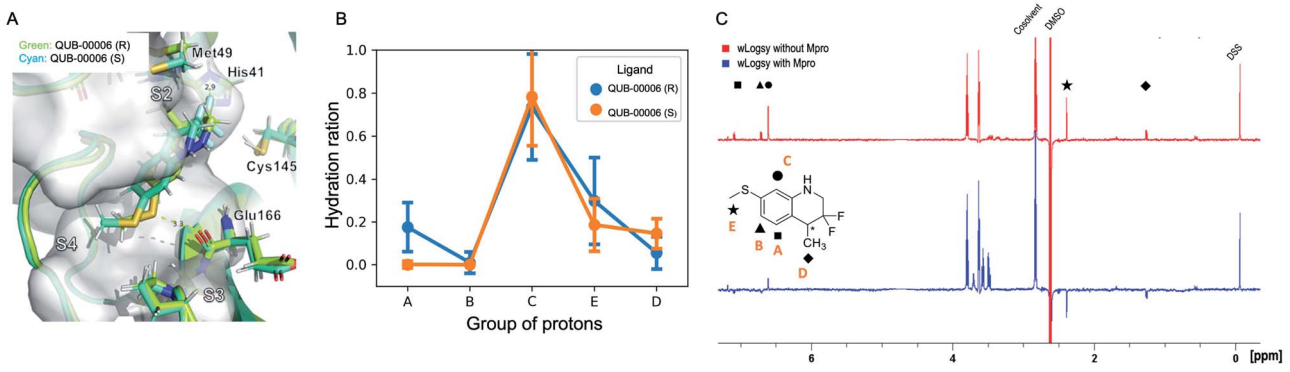


Fig. 3 Computational and experimental characterization of QUB-00006 binding within the M^{pro} binding pocket. (A) QUB-00006(*R*) (in light green) and QUB-00006(*S*) (in cyan) binding in a similar fashion at the interface of subpockets S2 and S4; the binding poses shown here were clustered and extracted from the trajectories of the binding free energy calculations performed on QUB-00006(*R*) and QUB-00006(*S*). (B) The analysis of our binding free energy trajectories showing that protons in groups A, B, E, and D have a low hydration ratio (less than 0.5), while the proton of group C has a high hydration ratio of 0.8. Hydration ratios calculated for the different proton groups of QUB-00006(*R*) correlate with those calculated for QUB-00006(*S*). (C) The WaterLOGSY spectra of QUB-00006 in the presence and absence of the M^{pro}. The assignment scheme is reported along with the 2D structure of the ligand. The strong negative intensity of the signals of the hydrogens of groups A, D, and E suggests that they are orientated towards the protein, while the hydrogen atom in C is solvent exposed. These experimental findings confirm the hydration ratio calculated during our binding free energy simulations and described in panel B.

E) and the methyl group at position 4 of the tetrahydroquinoline core (group D) are buried (Fig. 3B).

Those findings strongly correlate with the NMR characterization of QUB-00006 obtained *via* WaterLOGSY experiments. In fact, WaterLOGSY epitope mapping confirms that QUB-00006 binds to the protein binding pocket. We leveraged the experimental approach to better identify the region of the ligand in contact with the protein. In Fig. 3C, the proton signals arising from the two methyl groups (D and E) in the presence of M^{pro} show a change in the sign suggesting that these protons are in close contact with the protein. Similarly, the aromatic protons A and B undergo a sign inversion. In contrast, the aromatic proton C is not significantly perturbed, which suggests that this position is solvent exposed. The binding mode suggested by NMR is in agreement with the MD-derived hydration ratios confirming the predictive power of our MD-based approach to characterize the binding mode of novel ligands at the experimental level of accuracy (Fig. 3B and C).

Although we were able to gather structural information about the binding mode of QUB-00006 using a WaterLOGSY assay, we could not measure its experimental binding affinity *via* STD NMR due to solubility challenges.

Several synthetic steps were performed in order to obtain QUB-00006, as detailed in Fig. 4. Through this synthetic scheme, we obtained different intermediates characterized by a better solubility profile (Table 2). Interestingly, the hydroxyquinolinone QUB-00006-Int-01 displayed the best solubility profile of all the synthetic intermediates, making it a strong candidate for *in vitro* evaluation.

Before conducting NMR STD experiments to determine the dissociation constant (K_d) of the more polar QUB-00006-Int-01 compound, we decided to predict its binding conformation at the binding pocket and compute the respective absolute binding free energy. Modification of the molecular scaffolds, especially in fragment-like molecules, might affect the binding mode[74] compared to a reference structure (*e.g.* x0195 as per PDB ID:5R81).

Table 2 Prediction of the properties of compounds using our machine learning workflow. MW represents the molecular weight of the compounds in daltons, log *S* is the predicted solubility of the different compounds, log *P* represents the differential solubility, and the Tanimoto coefficient reflects the similarity of the selected compounds relative to x0195

	MW (Da)	log *S*	log *P*	Tanimoto (MACCS)
QUB-00 006	229.07	−3.99	3.56	0.391
QUB-00 006-Int-07	243.02	−3.73	1.96	0.371
QUB-00 006-Int-01	245.03	−2.73	1.66	0.338
x0195	226.08	−1.94	0.56	1

We used a combination of docking, MD and ABFE calculations to explore the putative binding mode of QUB-00006-Int-01. Those calculations identified two dominant binding modes for QUB-00006-Int-01(*R*) and QUB-00006-Int-01(*S*) (Fig. 5A) with computed binding free energies of −4.4 and −4.3 kcal mol^{-1}, respectively. Then, we estimated the binding affinity of QUB-00006-Int-01 towards M^{pro} by a STD NMR titration and we found a dissociation constant in the low millimolar range, with an estimated K_d of 1.9 ± 0.6 mM (−3.71 ± 0.2 kcal mol^{-1}), which agrees reasonably well with our binding free energy calculations (Table 1). As shown in Fig. 5A, both enantiomers bind to the S2 and S4 subpockets with the thioether group being fully buried

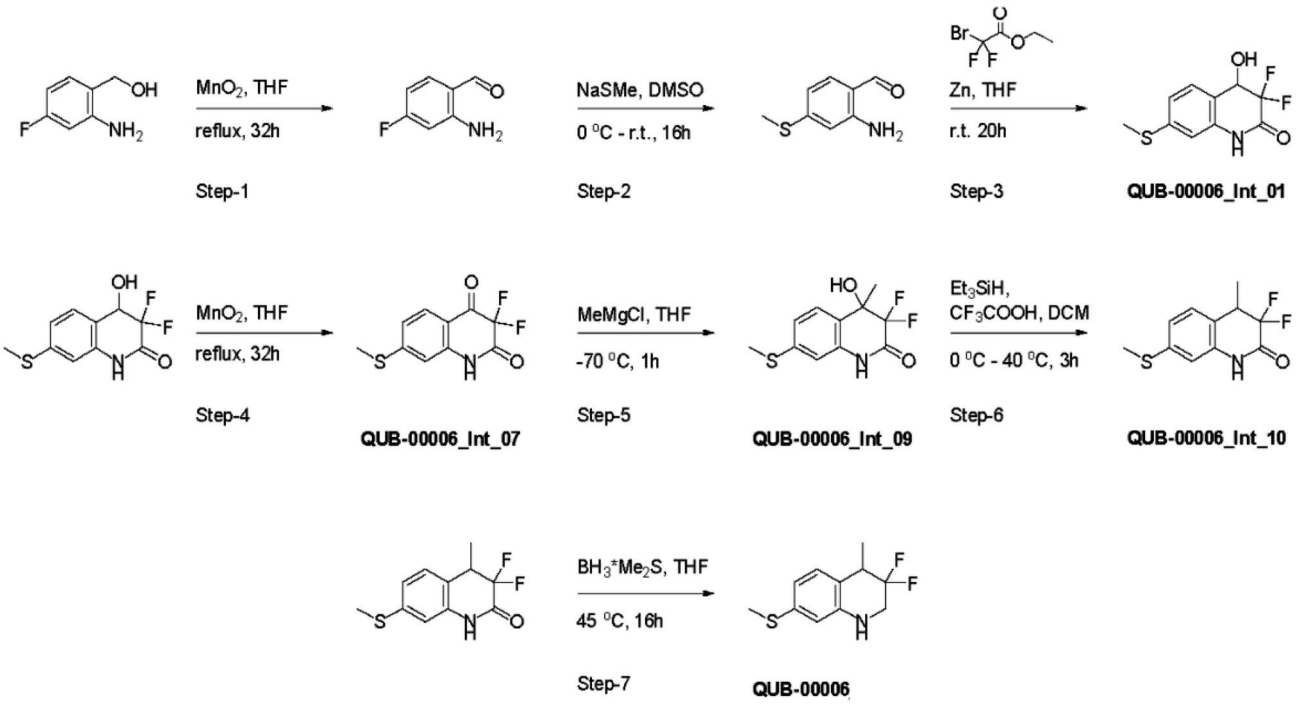


Fig. 4 Synthesis path of 3,3-difluoro-4-methyl-7-(methylsulfanyl)-1,2,3,4-tetrahydroquinoline named QUB-00006.

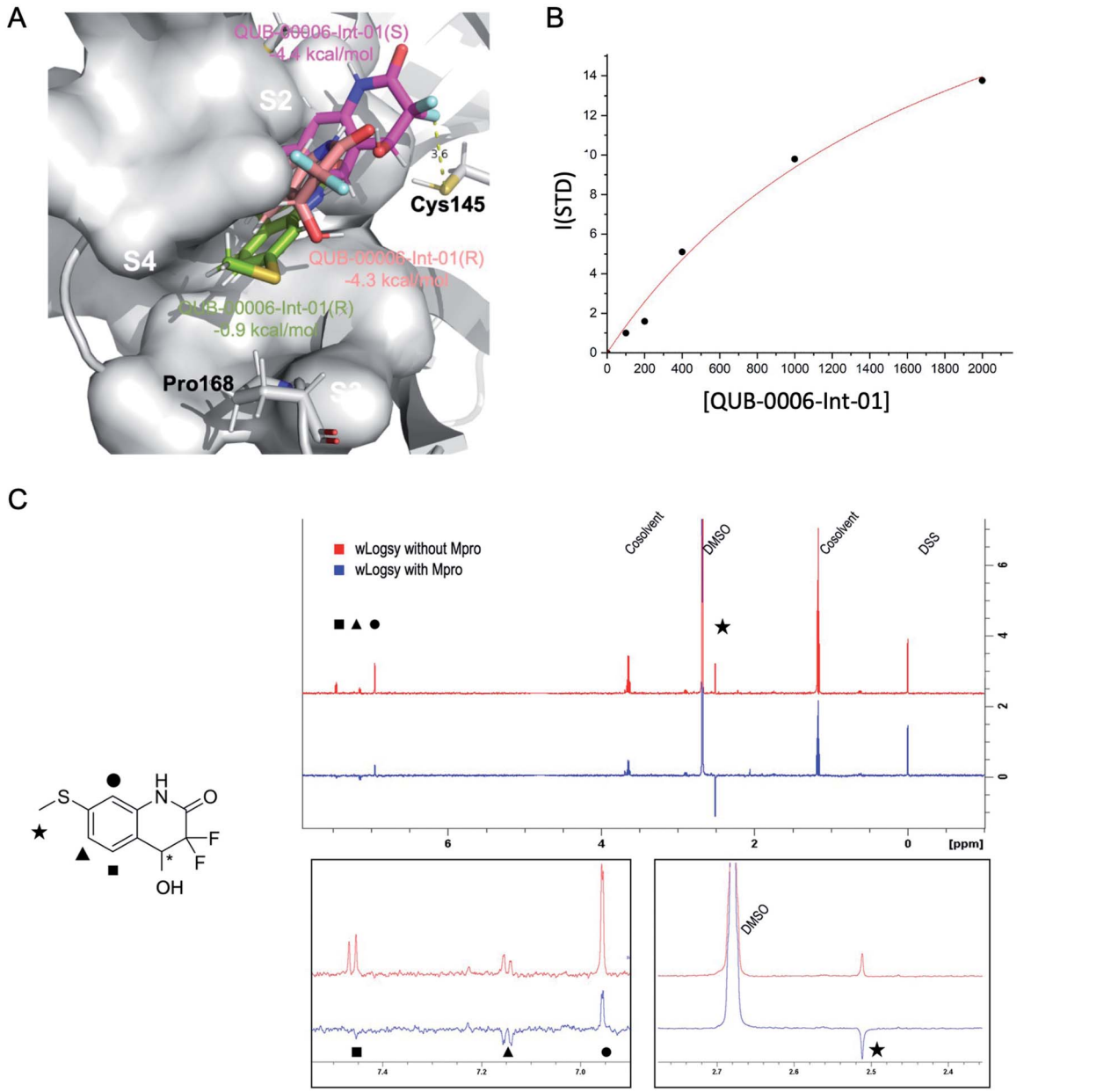


Fig. 5 Computational and experimental characterization of QUB-00006-Int-01 in the M^{pro} binding pocket. (A) The dominant binding modes of QUB-00006-Int-01(*R*) (in pink) and QUB-00006-Int-01(*S*) (in magenta), identified during ABFE simulations. They have computed binding free energies of −4.4 and −4.3 kcal mol^{-1}, respectively; also, they bind to the S2 and S4 subpockets in a similar fashion with the thioether group being fully buried in S2. On the other hand, starting with a QUB-00006 like binding mode, we ran an additional absolute binding free energy calculation on M^{pro} in complex with QUB-00006-Int-01(*R*) and obtained a second binding mode for QUB-00006-Int-01(*R*) (in green) with a binding free energy of −0.9 kcal mol^{-1}. (B) STD titration profile of QUB-00006-Int-01. The ligand concentration ranges from 100 μM to 2 mM against 10 μM of M^{pro}. (C) The WaterLOGSY spectra of QUB-00006-Int-01 with M^{pro} (in blue) and without M^{pro} (in red). The assignment of the signals is reported on the 2D structure of the fragment. The methyl and the aromatic signals of the two protons adjacent to the hydroxyl group undergo a significant change, which suggests that these groups are in close contact with the protein's cavity. In contrast, the aromatic proton adjacent to the lactamic nitrogen undergoes a reduction of its intensity, suggesting that this proton is partially exposed to the solvent. These STD results confirm our computational characterization of the binding mode of QUB-00006-Int-01 (panel A).

in subpocket S2, which correlates with WaterLOGSY experiments (Fig. 5C). Additionally, QUB-00006-Int-01(*R*) and QUB-00006-Int-01(*S*) fill up a binding pocket space that is different from the one occupied by QUB-00006. On the other hand, starting with a QUB-00006-like binding mode, we ran an additional absolute binding free energy calculation on an M^{pro}–QUB-00006-Int-01(*R*) complex and obtained a binding free energy of −0.9 kcal mol^{-1}. These results suggest that QUB-00006-Int-01 and QUB-00006 might have different dominant binding conformations (see Fig. 3A and 5A).

Since a fragment-like molecule could have multiple binding modes and the ligand conformation is unlikely to be fully

Table 3 Population of the clusters generated by adaptive sampling performed on M^{pro} in complex with QUB-00 006-Int-01(*R*) and (*S*). $\Delta\Delta G$ (kcal mol^{-1}) is the relative free energy at 298 K. The relative binding free energies reported for QUB-00 006-Int-01(*R*) and (*S*) are calculated using the respective cluster 1 as a reference ligand

QUB-00 006-Int-01(*R*)			QUB-00 006-Int-01(*S*)		
Cluster	Fraction	$\Delta\Delta G$	Cluster	Fraction	$\Delta\Delta G$
1	0.101	0	1	0.103	0
2	0.083	0.05	2	0.093	0.03
3	0.067	0.11	3	0.088	0.04
4	0.053	0.17	4	0.065	0.12
5	0.042	0.23	5	0.059	0.14
6	0.035	0.27	6	0.054	0.17
7	0.034	0.28	7	0.049	0.19
8	0.033	0.29	8	0.039	0.25
9	0.032	0.30	9	0.033	0.29
10	0.032	0.30	10	0.031	0.31

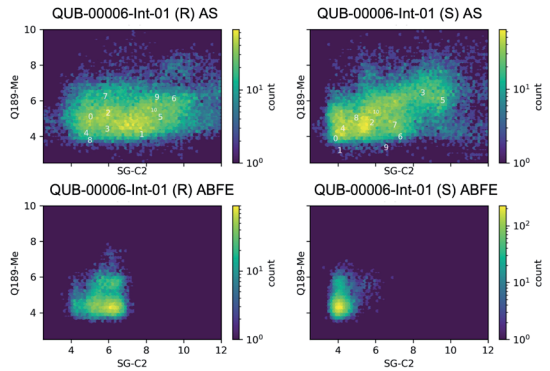


Fig. 6 Conformations of QUB-00006-Int-01 sampled during 20 ns of ABFE calculations and 450 ns of adaptive sampling simulations. The conformation was plotted as a function of two distances: (i) the distance between C2 (carbon of QUB-00006-Int-01 connected to the hydroxyl group) and the sulfur of Cys145, and (ii) the distance between the methyl thioether group in QUB-00006-Int-01 and the beta carbon of Gln189. "0" indicates the starting structure, "1" indicates the largest cluster, and "*i*" indicates the *i*th largest cluster. The frames were taken at 10 ps time intervals.

sampled during 20 ns of binding free energy simulations, we used unsupervised adaptive sampling (AS) to further explore the conformational space of QUB-00006-Int-01. AS can be used here as an interpretative tool able to gather structural insights on the various potential M^{pro}–ligand interactions (see the ESI‡ for details). The AS trajectories were clustered using average-linkage hierarchical clustering algorithms and the top ten largest clusters were chosen for analysis. These clusters have comparable populations (the smallest clusters have 3–4 times smaller populations or 0.3 kcal mol^{-1} higher free energy than the largest clusters, see Table 3), indicating the coexistence of multiple binding modes.

More precisely, starting from these clusters, absolute binding free energies would yield results within 0.3 kcal mol^{-1} of what was previously obtained. The simulations of QUB-00006-Int-01(*R*) and QUB-00006-Int-01(*S*) converged to similar ensembles containing several possible binding modes. Clusters 3, 5, and 6 of QUB-00006-Int-01(*R*) and cluster 4 of QUB-00006-Int-01(*S*) (ESI-Fig. 2‡) correspond to the respective dominant binding modes predicted by ABFE simulations (Fig. 5A). For both enantiomers, the most conserved interactions are the hydrophobic contacts between C9 (methyl thioether) and Gln189, and between C5 (proton B) and His41, Arg188, and Gln189.

Overall, our computational findings on QUB-00006-Int-01 confirm that the structural approach we introduce in this work using a sequence of MD-based techniques (classical MD simulations, adaptive sampling, and absolute binding free energy calculations) is able to capture potential binding orientations of fragment-like compounds in the binding pocket of a protein, and to accurately predict their binding free energies.

Then, we analyzed the clustered QUB-00006-Int-01 binding conformations from the adaptive sampling simulations plotted as a function of the distance between the methyl thioether group in QUB-00006-Int-01 and the beta carbon of Gln189, and the distance between C2 (carbon connected to the hydroxyl group) and the sulfur atom (SG) of the catalytic side chain of the Cys145 residue (Fig. 6). We noticed that the distance of C2–SG in the most populated cluster generated by the AS simulations is around 4 Å. To reinforce our analysis, we leveraged another unsupervised reduction of dimension technique: TICA (time-lagged independent component analysis),[75] which aims at finding the slow collective variables of the data, and applied it to QUB-00006-Int-01(*R*). We then used the *k*-means clustering method on the data projected on this space and built a Hidden Markov State Model (HMSM).[76] Three clusters emerged, whose characteristics also show the coexistence of several binding modes of QUB-00006-Int-01(*R*), one of which corresponds to a distance between C2 and SG below 4 Å. Detailed results can be found in the ESI.‡

Targeting Cys145 with covalent warheads has been used by several researchers to discover novel potent inhibitors of M^{pro}.[38,63,77] As a matter of fact, a simple chemical modification to QUB-00006-Int-01 would lead to QUB-00006-Int-07 bearing an α,α-difluoro-keto moiety, which is prone to a nucleophilic attack by the vicinal R-SH of Cys145. In order to enable the latter, QUB-00006-Int-07 would need to access the M^{pro} substrate pocket and adopt a stable binding conformation prior to the covalent binding. Thus, we conducted absolute binding free energy simulations on the M^{pro}–QUB-00006-Int-07 complex, which confirmed a favorable binding energy of QUB-00006-Int-07 to the M^{pro} substrate pocket (-5.37 ± 0.23 kcal mol^{-1}). As reported in Fig. 7, compound QUB-00006-Int-07 is bound to the S2 and S4 subpockets with the thioether group being fully buried in subpocket S2 and the α,α-difluoro-keto moiety facing Cys145. More precisely, the average distance between SG and the C is 3.65 angstroms (±0.33) and the average distance between the SG and C2 is 3.61 angstroms (±0.43) as can be seen in Fig. 7.

Our computational findings motivated us to test the compound with a FRET-based proteolytic assay. This assay should detect potent functional binders to the viral M^{pro}. Being a fluorogenic assay, compounds with fluorescence quenching properties can suppress the fluorescence signal generated by

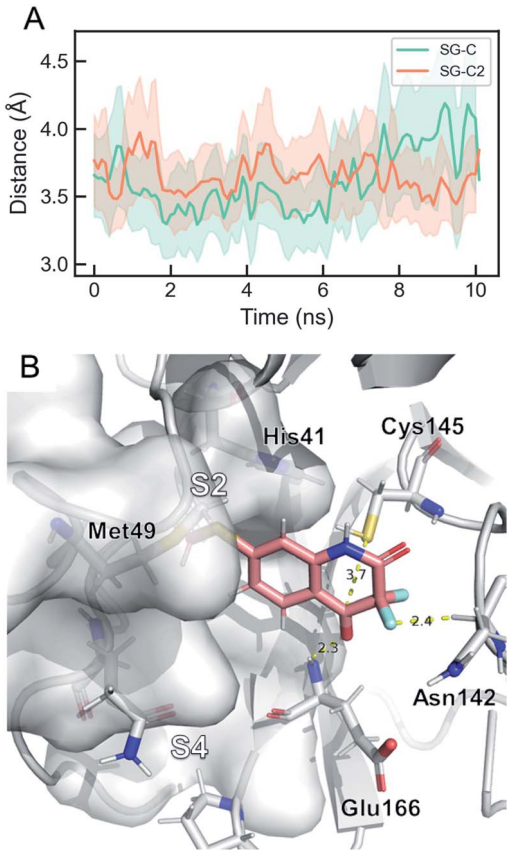


Fig. 7 The dominant binding mode of QUB-00006-Int-07 during ABFE simulations. (A) Time evolution of key distances in the simulation. "SG" stands for the sulfur atom in Cys145, "C" is the amide carbon in QUB-00006-Int-07, and "C2" is the carbonyl carbon in QUB-00006-Int-07. The average distances for SG–C and SG–C2 are 3.61 Å and 3.65 Å, respectively. (B) The dominant binding mode of QUB-00006-Int-07 within the M^{pro} binding pocket. QUB-00006-Int-07 is shown in pink and the protein is shown in silver sticks and surfaces. The binding mode is very stable during the simulation, where the hydroxyl group is close to Cys145 and forms a hydrogen-bond with the Glu166 backbone, and the difluoro group interacts with the carbonyl group of Asn142. These binding modes are also comparable to the dominant binding mode of QUB-00006-Int-01(*S*) identified during ABFE calculations.

the protease activity. To eliminate false positive results, we conducted a preliminary counter screen and verified that the tested compound possesses negligible fluorescence quenching effects. Subsequently, to assess the potential inhibitory activity of the compound against SARS-CoV-2 M^{pro}, increasing concentrations of QUB-00006-Int-07 (0.25–150 μM) were incubated with 20 nM M^{pro} before the addition of 5 μM FRET substrate. As shown in Fig. 8, QUB-00006-Int-07 inhibited M^{pro} with 50% inhibitory concentration (IC_{50} value of 830 ± 50 nM), thus resulting in a fairly potent inhibitor of the M^{pro} enzymatic activity. The binding of QUB-00006-Int-07 to M^{pro} was confirmed by electrospray ionization (ESI) mass spectrometry.

A preliminary determination of the initial protein showed an experimental mass of 33 796.40 Da, which matches very closely the expected value of 33 796.64 Da calculated from the sequence

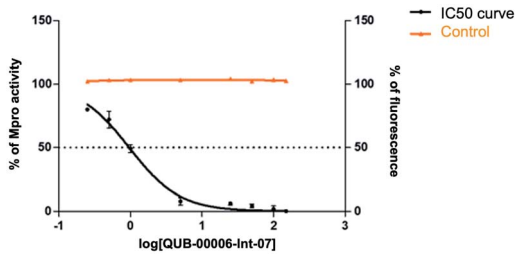


Fig. 8 Dose–response curves obtained by plotting the percentage of SARS-CoV-2 M^{pro} residual activity as a function of increasing concentrations of QUB-00006-Int-07 (0–150 μM). [M^{pro}] = 20 nM, [PS1] = 5 μM, %DMSO = 3.75%. Experiments were performed in triplicate. A counter screening control experiment was performed by testing increasing concentrations of QUB-00006-Int-07 in the presence of 0.5 μM free fluorescein.

(Fig. 9A). The sample obtained after incubation of QUB-00006-Int-07 with M^{pro} (compound : protein ratio = 10 : 1) was analyzed by ESI-MS under denaturing conditions, and a representative spectrum is provided in Fig. 9B. In addition to the signals corresponding to multiple charge states of the initial protein (red dots), we identified the distribution of signals corresponding to the M^{pro} modified by the presence of the compound (green asterisks) which is therefore covalently linked to the protein given the non-native conditions of the experiment. The nature of the adduct and the molecular mechanism of binding are under investigation and will be the subject of further studies.

Finally, in this work, the introduction of multiple modifications (*e.g. gem*-difluoro, thioether, hydroxyl and methyl groups) to the tetrahydroquinoline scaffold of x0195, and the design and synthesis of novel molecular scaffolds, enabled the exploration of binding pocket boundaries and provided additional information related to druggability of the S2 subpocket. Other

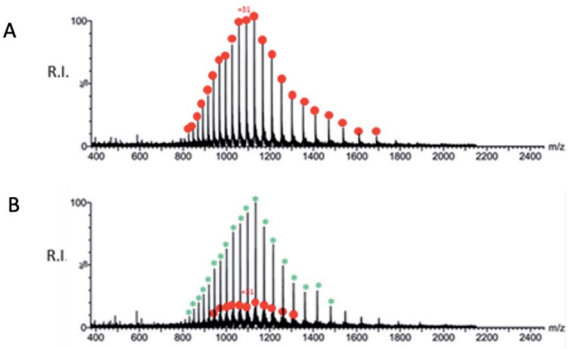


Fig. 9 (A) Representative ESI-MS spectrum of a solution containing 10 μM SARS-CoV-2 M^{pro} in water/acetonitrile (50 : 50) added with 0.1% formic acid. The spectrum was acquired in positive ion mode. (B) Representative ESI-MS spectrum of a mixture containing 10 μM SARS-CoV-2 M^{pro} after incubation with QUB-00006-Int-07 (compound : protein ratio = 10 : 1) in water/acetonitrile (50 : 50) added with 0.1% formic acid. The spectrum was acquired in positive ion mode. The red dots correspond to the unmodified protein, and the green asterisks correspond to the modified protein.

molecules were produced over the course of this research but, due to their weaker activity, their detailed analysis is not provided here. Their list can be found in the ESI.‡ These compounds were either designed computationally without leading to improved affinities or were synthesis intermediates. All resulting molecules were submitted to biological testing, but none of them were found to be as potent as QUB-00006-Int-07 nor presented a strongly druggable profile, compared to the previously discussed compounds.

5. Conclusion and perspectives

We presented a computationally driven discovery of a new set of non-covalent and covalent inhibitors of M^{pro} that have been further characterized experimentally. The best compound, QUB-00006-Int-07, has been found to be a covalent binder that resulted in a potent inhibition of the M^{pro} enzymatic activity ($IC_{50} = 830 \pm 50$ nM). The results of the innovative scaffold design described here were obtained within three months *via* a fast-track project that took place in the summer of 2021. It involved a small consortium of theoreticians, organic chemists and drug designers, and demonstrated the effectiveness of a computation-guided synthetic strategy. Indeed, GPU-accelerated high-performance computing platforms can now provide access to high-resolution molecular dynamics simulations, which are able to predict detailed protein conformational maps and provide accurate absolute binding free energy results. Such computations can be further rationalized by means of adaptive sampling simulations, an approach which is able to decipher multiple binding modes. Coupled to NMR, *in vitro* experiments and machine learning, such high-resolution predictions yield structural insights regarding the design of new active compounds, while offering an atomic level understanding of binding affinities.

Beyond this preliminary proof of concept study, the next research steps will be devoted to the QM/MM modeling[78,79] of the warhead reaction mechanism[38,77,80] leading to the covalent binding of QUB-00006-Int-07, and to optimization of active compounds with the goal of reaching low nanomolar activity.

Data availability

All data have been provided in the main text and ESI.†

Author contributions

L. E. K., Z. J., D. L., T. J. I., D. E. A., B. S., and L. L. performed the simulations. L. L., T. J. I., D. S., and J.-P. P. contributed new code. M. S., M. B., A. S., and M. V. performed NMR and mass spectroscopy experiments. I. K., R. I., A. K., Y. M., A. P., and V. C.-R. performed the synthesis. L. L., B. S., P. R., J. W. P., D. S., and J.-P. P. contributed new methodology. L. L., B. S., F. H., P. R., J. W. P., and J.-P. P. contributed analytical tools. All authors analyzed the data. L. E. K., Z. J., L. L., M. S., A. S., D. S. and J.-P. P. wrote the paper. D. S. and J.-P. P. designed the research.

Conflicts of interest

P. R., M. M., L. L., J. W. P. and J.-P. P. are co-founders and shareholders of Qubit Pharmaceuticals.

Acknowledgements

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 810367), project EMC2 (JPP). Computations have been made possible thanks to special COVID-19 grants from: (i) Amazon Web Services (AWS) allowing access to the AWS Cloud supercomputing platform (JPP; Qubit Pharmaceuticals); (ii) GENCI on the Jean Zay supercomputer (IDRIS, Orsay, France) through projects AP010712339 and AD011012316 (Qubit Pharmaceuticals); GENCI allocation no. A0070707671 (JPP). The work on experimental validation was supported by funding from the CARIPARO Foundation ("Progetti di ricerca sul COVID-19" No. 55812 to BG) and from the Department of Chemical Sciences (project P-DiSC 01BIRD2018-UNIPD to MB). DS thanks Denis Klapishevskiy for discussions, and Prof. Carsten Bolm and his team for providing the synthetic scheme for scaffolds related to the x0195 fragment (https://bolm.oc.rwth-aachen.de/content/outreach).

References

1 P. Zhou, X.-L. Yang, X.-G. Wang, B. Hu, L. Zhang, W. Zhang, H.-R. Si, Y. Zhu, B. Li, C.-L. Huang, *et al.*, *nature*, 2020, **579**, 270–273.
2 F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei, *et al.*, *Nature*, 2020, **579**, 265–269.
3 E. N. Muratov, R. Amaro, C. H. Andrade, N. Brown, S. Ekins, D. Fourches, O. Isayev, D. Kozakov, J. L. Medina-Franco, K. M. Merz, T. I. Oprea, V. Poroikov, G. Schneider, M. H. Todd, A. Varnek, D. A. Winkler, A. V. Zakharov, A. Cherkasov and A. Tropsha, *Chem. Soc. Rev.*, 2021, **50**, 9121–9151.
4 F. von Delft, M. Calmiano, J. Chodera, E. Griffen, A. Lee, N. London, T. Matviuk, B. Perry, M. Robinson and A. von Delft, *A white-knuckle ride of open COVID drug discovery*, 2021.
5 J. Breidenbach, C. Lemke, T. Pillaiyar, L. Schäkel, G. Al Hamwi, M. Diett, R. Gedschold, N. Geiger, V. Lopez, S. Mirza, V. Namasivayam, A. C. Schiedel, K. Sylvester, D. Thimm, C. Vielmuth, L. Phuong Vu, M. Zyulina, J. Bodem, M. Gütschow and C. E. Müller, *Angew. Chem., Int. Ed.*, 2021, **60**, 10423–10429.
6 F. Krammer, *Nature*, 2020, **586**, 516–527.
7 H. Ledford, *et al.*, *Nature*, 2021, **599**, 358–359.
8 L. Zhang, D. Lin, Y. Kusov, Y. Nian, Q. Ma, J. Wang, A. von Brunn, P. Leyssen, K. Lanko, J. Neyts, A. de Wilde, E. J. Snijder, H. Liu and R. Hilgenfeld, *J. Med. Chem.*, 2020, **63**, 4562–4578.
9 H. Yang, W. Xie, X. Xue, K. Yang, J. Ma, W. Liang, Q. Zhao, Z. Zhou, D. Pei, J. Ziebuhr, *et al.*, *PLoS Biol.*, 2005, **3**, e324.

10 Z. Cournia, B. Allen and W. Sherman, *J. Chem. Inf. Model.*, 2017, **57**, 2911–2937.
11 D. W. Kneller, G. Phillips, H. M. O'Neill, R. Jedrzejczak, L. Stols, P. Langan, A. Joachimiak, L. Coates and A. Kovalevsky, *Nat. Commun.*, 2020, **11**, 1–6.
12 T. Jaffrelot Inizan, F. Célerse, O. Adjoua, D. El Ahdab, L.-H. Jolly, C. Liu, P. Ren, M. Montes, N. Lagarde, L. Lagardère, P. Monmarché and J.-P. Piquemal, *Chem. Sci.*, 2021, **12**, 4889–4907.
13 D. El Ahdab, L. Lagardère, T. J. Inizan, F. Célerse, C. Liu, O. Adjoua, L.-H. Jolly, N. Gresh, Z. Hobaika, P. Ren, R. G. Maroun and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2021, **12**, 6218–6226.
14 S. Zev, K. Raz, R. Schwartz, R. Tarabeh, P. K. Gupta and D. T. Major, *J. Chem. Inf. Model.*, 2021, **61**, 2957–2966.
15 Z. Li, X. Li, Y.-Y. Huang, Y. Wu, R. Liu, L. Zhou, Y. Lin, D. Wu, L. Zhang, H. Liu, X. Xu, K. Yu, Y. Zhang, J. Cui, C.-G. Zhan, X. Wang and H.-B. Luo, *Proc. Natl. Acad. Sci. U. S. A.*, 2020, **117**, 27381–27387.
16 H. T. H. Chan, M. A. Moesser, R. K. Walters, T. R. Malla, R. M. Twidale, T. John, H. M. Deeks, T. Johnston-Wood, V. Mikhailov, R. B. Sessions, W. Dawson, E. Salah, P. Lukacik, C. Strain-Damerell, C. D. Owen, T. Nakajima, K. Świderek, A. Lodola, V. Moliner, D. R. Glowacki, J. Spencer, M. A. Walsh, C. J. Schofield, L. Genovese, D. K. Shoemark, A. J. Mulholland, F. Duarte and G. M. Morris, *Chem. Sci.*, 2021, **12**, 13686–13703.
17 A. Morris, W. McCorkindale, T. C. M. Consortium, N. Drayman, J. D. Chodera, S. Tay, N. London and A. A. Lee, *Chem. Commun.*, 2021, **57**, 5909–5912.
18 D. Shcherbakov, D. Baev, M. Kalinin, A. Dalinger, V. Chirkova, S. Belenkaya, A. Khvostov, D. Krut'ko, A. Medved'ko, E. Volosnikova, E. Sharlaeva, D. Shanshin, T. Tolstikova, O. Yarovaya, R. Maksyutov, N. Salakhutdinov and S. Vatsadze, *ACS Med. Chem. Lett.*, 2021, **13**(1), 140–147.
19 E. Glaab, G. B. Manoharan and D. Abankwa, *J. Chem. Inf. Model.*, 2021, **61**, 4082–4096.
20 Z. Jin, X. Du, Y. Xu, Y. Deng, M. Liu, Y. Zhao, B. Zhang, X. Li, L. Zhang, C. Peng, Y. Duan, J. Yu, L. Wang, K. Yang, F. Liu, R. Jiang, X. Yang, T. You, X. Liu, X. Yang, F. Bai, H. Liu, X. Liu, L. W. Guddat, W. Xu, G. Xiao, C. Qin, Z. Shi, H. Jiang, Z. Rao and H. Yang, *Nature*, 2020, **582**, 289–293.
21 J. Gossen, S. Albani, A. Hanke, B. P. Joseph, C. Bergh, M. Kuzikov, E. Costanzi, C. Manelfi, P. Storici, P. Gribbon, A. R. Beccari, C. Talarico, F. Spyrakis, E. Lindahl, A. Zaliani, P. Carloni, R. C. Wade, F. Musiani, D. B. Kokh and G. Rossetti, *ACS Pharmacol. Transl. Sci.*, 2021, **4**, 1079–1095.
22 A. Manandhar, V. Srinivasulu, M. Hamad, H. Tarazi, H. Omar, D. J. Colussi, J. Gordon, W. Childers, M. L. Klein, T. H. Al-Tel, M. Abou-Gharbia and K. M. Elokely, *J. Chem. Inf. Model.*, 2021, **61**, 4745–4757.
23 G. Amendola, R. Ettari, S. Previti, C. Di Chio, A. Messere, S. Di Maro, S. J. Hammerschmidt, C. Zimmer, R. A. Zimmermann, T. Schirmeister, M. ZappalÃ and S. Cosconati, *J. Chem. Inf. Model.*, 2021, **61**, 2062–2073.
24 M. M. Ghahremanpour, J. Tirado-Rives, M. Deshmukh, J. A. Ippolito, C.-H. Zhang, I. Cabeza de Vaca, M.-E. Liosi, K. S. Anderson and W. L. Jorgensen, *ACS Med. Chem. Lett.*, 2020, **11**, 2526–2533.
25 C.-H. Zhang, K. A. Spasov, R. A. Reilly, K. Hollander, E. A. Stone, J. A. Ippolito, M.-E. Liosi, M. G. Deshmukh, J. Tirado-Rives, S. Zhang, Z. Liang, S. J. Miller, F. Isaacs, B. D. Lindenbach, K. S. Anderson and W. L. Jorgensen, *ACS Med. Chem. Lett.*, 2021, **12**, 1325–1332.
26 C.-H. Zhang, E. A. Stone, M. Deshmukh, J. A. Ippolito, M. M. Ghahremanpour, J. Tirado-Rives, K. A. Spasov, S. Zhang, Y. Takeo, S. N. Kudalkar, Z. Liang, F. Isaacs, B. Lindenbach, S. J. Miller, K. S. Anderson and W. L. Jorgensen, *ACS Cent. Sci.*, 2021, **7**, 467–475.
27 O. Adjoua, L. Lagardère, L.-H. Jolly, A. Durocher, T. Very, I. Dupays, Z. Wang, T. J. Inizan, F. Célerse, P. Ren, J. W. Ponder, J.-P. Piquemal, *et al.*, *J. Chem. Theory Comput.*, 2021, **17**, 2034–2053.
28 L. Lagardère, L.-H. Jolly, F. Lipparini, F. Aviat, B. Stamm, Z. F. Jing, M. Harger, H. Torabifard, G. A. Cisneros, M. J. Schnieders, N. Gresh, Y. Maday, P. Y. Ren, J. W. Ponder and J.-P. Piquemal, *Chem. Sci.*, 2018, **9**, 956–972.
29 P. Ren and J. W. Ponder, *J. Phys. Chem. B*, 2003, **107**, 5933–5947.
30 J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. A. DiStasio Jr, *et al.*, *J. Phys. Chem. B*, 2010, **114**, 2549–2564.
31 Y. Shi, Z. Xia, J. Zhang, R. Best, C. Wu, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2013, **9**, 4046–4063.
32 C. Zhang, C. Lu, Z. Jing, C. Wu, J.-P. Piquemal, J. W. Ponder and P. Ren, *J. Chem. Theory Comput.*, 2018, **14**, 2084–2108.
33 D. Jiao, P. A. Golubkov, T. A. Darden and P. Ren, *Proc. Natl. Acad. Sci. U. S. A.*, 2008, **105**, 6290–6295.
34 Y. Shi, C. Z. Zhu, S. F. Martin and P. Ren, *J. Phys. Chem. B*, 2012, **116**, 1716–1727.
35 R. Qi, Z. Jing, C. Liu, J.-P. Piquemal, K. N. Dalby and P. Ren, *J. Phys. Chem. B*, 2018, **122**, 6371–6376.
36 Z. Jing, J. A. Rackers, L. R. Pratt, C. Liu, S. B. Rempe and P. Ren, *Chem. Sci.*, 2021, **12**, 8920–8930.
37 Y. Shi, M. L. Laury, Z. Wang and J. W. Ponder, *J. Comput.-Aided Mol. Des.*, 2021, **35**, 79–93.
38 K. Arafet, N. Serrano-Aparicio, A. Lodola, A. J. Mulholland, F. V. González, K. Świderek and V. Moliner, *Chem. Sci.*, 2021, **12**, 1433–1444.
39 L.-H. Jolly, A. Duran, L. Lagardère, J. W. Ponder, P. Ren and J.-P. Piquemal, *Living Journal of Computational Molecular Science*, 2019, **1**, 10409.
40 O. Trott and A. J. Olson, *J. Comput. Chem.*, 2010, **31**, 455–461.
41 D. Santos-Martins, L. Solis-Vasquez, A. F. Tillack, M. F. Sanner, A. Koch and S. Forli, *J. Chem. Theory Comput.*, 2021, **17**, 1060–1073.
42 J. A. Rackers, Z. Wang, C. Lu, M. L. Laury, L. Lagardère, M. J. Schnieders, J.-P. Piquemal, P. Ren and J. W. Ponder, *J. Chem. Theory Comput.*, 2018, **14**, 5273–5289.
43 U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee and L. G. Pedersen, *J. Chem. Phys.*, 1995, **103**, 8577–8593.

44 L. Lagardère, F. Lipparini, E. Polack, B. Stamm, E. Cancès, M. Schnieders, P. Ren, Y. Maday and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2015, **11**, 2589–2599.
45 L. Lagardère, F. Aviat and J.-P. Piquemal, *J. Phys. Chem. Lett.*, 2019, **10**, 2593–2599.
46 L. Lagardère, F. Lipparini, E. Polack, B. Stamm, E. Cancès, M. Schnieders, P. Ren, Y. Maday and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2015, **11**, 2589–2599.
47 G. M. Morris, R. Huey, W. Lindstrom, M. F. Sanner, R. K. Belew, D. S. Goodsell and A. J. Olson, *J. Comput. Chem.*, 2009, **30**, 2785–2791.
48 G. L. Warren, C. W. Andrews, A.-M. Capelli, B. Clarke, J. LaLonde, M. H. Lambert, M. Lindvall, N. Nevins, S. F. Semus, S. Senger, G. Tedesco, I. D. Wall, J. M. Woolven, C. E. Peishoff and M. S. Head, *J. Med. Chem.*, 2006, **49**, 5912–5931.
49 R. T. McGibbon, K. A. Beauchamp, M. P. Harrigan, C. Klein, J. M. Swails, C. X. Hernández, C. R. Schwantes, L.-P. Wang, T. J. Lane and V. S. Pande, *Biophys. J.*, 2015, **109**, 1528–1532.
50 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, *Journal of Machine Learning Research*, 2011, **12**, 2825–2830.
51 C. H. Bennett, *J. Comput. Phys.*, 1976, **22**, 245–268.
52 M. C. Sorkun, A. Khetan and S. Er, *Sci. Data*, 2019, **6**, 143.
53 K. Mansouri, C. M. Grulke, R. S. Judson and A. J. Williams, *J. Cheminf.*, 2018, **10**, 10.
54 D. Bajusz, A. Rácz and K. Héberger, *J. Cheminf.*, 2015, **7**, 1–13.
55 H. L. Morgan, *J. Chem. Doc.*, 1965, **5**, 107–113.
56 L. Zhang, D. Lin, X. Sun, U. Curth, C. Drosten, L. Sauerhering, S. Becker, K. Rox and R. Hilgenfeld, *Science*, 2020, **368**, 409–412.
57 Y. Liu, W. Kati, C.-M. Chen, R. Tripathi, A. Molla and W. Kohlbrenner, *Anal. Biochem.*, 1999, **267**, 331–335.
58 C. Ma, M. D. Sacco, B. Hurst, J. A. Townsend, Y. Hu, T. Szeto, X. Zhang, B. Tarbet, M. T. Marty, Y. Chen and J. Wang, *Cell Res.*, 2020, **30**, 678–692.
59 C. Dalvit, G. Fogliatto, A. Stewart, M. Veronesi and B. Stockman, *J. Biomol. NMR*, 2001, **21**, 349–359.
60 A. Mier, I. Maffucci, F. Merlier, E. Prost, V. Montagna, G. U. Ruiz-Esparza, J. V. Bonventre, P. K. Dhal, B. Tse Sum Bui, P. Sakhaii and K. Haupt, *Angew. Chem., Int. Ed.*, 2021, **60**, 20849–20857.
61 A. Douangamath, D. Fearon, P. Gehrtz, T. Krojer, P. Lukacik, C. D. Owen, E. Resnick, C. Strain-Damerell, A. Aimon, P. Ábrányi Balogh and others, *Nat. Commun.*, 2020, **11**, 1–11.
62 A. L. Kantsadi, E. Cattermole, M.-T. Matsoukas, G. A. Spyroulias and I. Vakonakis, *J. Biomol. NMR*, 2021, **75**, 167–178.
63 R. Cannalire, C. Cerchia, A. R. Beccari, F. S. Di Leva and V. Summa, *J. Med. Chem.*, 2020, DOI: 10.1021/acs.jmedchem.0c01140.
64 K. A. Brameld, B. Kuhn, D. C. Reuter and M. Stahl, *J. Chem. Inf. Model.*, 2008, **48**, 1–24.
65 J.-D. Chai and M. Head-Gordon, *Phys. Chem. Chem. Phys.*, 2008, **10**, 6615–6620.
66 V. A. Rassolov, J. A. Pople, M. A. Ratner and T. L. Windus, *J. Chem. Phys.*, 1998, **109**, 1223–1229.
67 M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery Jr, J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman and D. J. Fox, *Gaussian ~16 Revision C.01*, Gaussian Inc, Wallingford CT, 2016.
68 M. Iwaoka, S. Takemoto and S. Tomoda, *J. Am. Chem. Soc.*, 2002, **124**, 10613–10620.
69 S. Kortagere, S. Ekins and W. J. Welsh, *J. Mol. Graphics Modell.*, 2008, **27**, 170–177.
70 M. R. Bauer, R. N. Jones, M. G. J. Baud, R. Wilcken, F. M. Boeckler, A. R. Fersht, A. C. Joerger and J. Spencer, *ACS Chem. Biol.*, 2016, **11**, 2265–2274.
71 K. El Hage, J.-P. Piquemal, Z. Hobaika, R. G. Maroun and N. Gresh, *J. Comput. Chem.*, 2013, **34**, 1125–1135.
72 X. Mu, Q. Wang, L.-P. Wang, S. D. Fried, J.-P. Piquemal, K. N. Dalby and P. Ren, *J. Phys. Chem. B*, 2014, **118**, 6456–6465.
73 J. Melcr and J.-P. Piquemal, *Front. Mol. Biosci.*, 2019, **6**, 143.
74 C. P. Mpamhanga, D. Spinks, L. B. Tulloch, E. J. Shanks, D. A. Robinson, I. T. Collie, A. H. Fairlamb, P. G. Wyatt, J. A. Frearson, W. N. Hunter, I. H. Gilbert and R. Brenk, *J. Med. Chem.*, 2009, **52**, 4454–4465.
75 G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis and F. Noé, *J. Chem. Phys.*, 2013, **139**, 07B604_1.
76 M. K. Scherer, B. Trendelkamp-Schroer, F. Paul, G. Pérez-Hernández, M. Hoffmann, N. Plattner, C. Wehmeyer, J.-H. Prinz and F. Noé, *J. Chem. Theory Comput.*, 2015, **11**, 5525–5542.
77 S. Martí, K. Arafet, A. Lodola, A. J. Mulholland, K. Świderek and V. Moliner, *ACS Catal.*, 2022, **12**, 698–708.
78 D. Loco, L. Lagardère, S. Caprasecca, F. Lipparini, B. Mennucci and J.-P. Piquemal, *J. Chem. Theory Comput.*, 2017, **13**, 4025–4033.
79 D. Loco, L. Lagardère, G. A. Cisneros, G. Scalmani, M. Frisch, F. Lipparini, B. Mennucci and J.-P. Piquemal, *Chem. Sci.*, 2019, **10**, 7200–7211.
80 K. Świderek and V. Moliner, *Chem. Sci.*, 2020, **11**, 10626–10630.

Conclusion

A computationally driven discovery of a new set of non-covalent and covalent inhibitors of M^{pro} that have been further characterized experimentally are presented here. The best compound has been found to be a covalent binder that resulted in a potent inhibition of the M^{pro} activity. Coupled to NMR, in vitro experiments and machine learning, such high-resolution predictions yield structural insights regarding the design of new active compounds. The deep learning-driven Hidden Markov State Models led to three relevant clusters. For each clusters, the relevant conformational structure extracted have characteristics that show the coexistence of several binding modes of the covalent binder.

The past research work show how the adaptive sampling algorithm can be used at many level to tackle global challenges such as ligand binding modes and protein conformational sampling. However, thanks to its general concept the algorithm can be coupled with many enhanced sampling techniques enabling larger conformational space of larger biomolecular systems. In the last section, we will discuss in detail a multi-level strategy that is coupling this adaptive sampling algorithm with a novel gaussian-accelerated molecular dynamics model designed for PFFs and umbrella sampling that were tested on the conformational space of a large protein.