Reduced order models in quasi-static nonlinear mechanics for state estimation by calibration through data assimilation : application to containment buildings

Eki Agouzal

To cite this version:

Eki Agouzal. Reduced order models in quasi-static nonlinear mechanics for state estimation by calibration through data assimilation : application to containment buildings. Mathematics [math]. Université de Bordeaux, 2024. English. NNT : 2024BORD0075 . tel-04681011

HAL Id: tel-04681011
https://theses.hal.science/tel-04681011v1

Submitted on 29 Aug 2024

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR
DE L'UNIVERSITÉ DE BORDEAUX

Ecole Doctorale de mathématiques et Informatique

Mathématiques appliquées et calcul scientifique

Par **Eki AGOUZAL**

Réduction de modèles en mécanique non-linéaire quasi-statique pour l'estimation de l'état par recalage en assimilation de données: application aux enceintes de confinement

Reduced order models in quasi-static nonlinear mechanics for state estimation by calibration through data assimilation: application to containment buildings

Sous la direction de : **Michel BERGMANN** et **Tommaso TADDEI**

Soutenue le 25 avril 2024

Membres du jury :

Mme. Virginie EHRLACHER	Professeure	Ecole des Ponts ParisTech	Rapporteur
M. Ludovic CHAMOIN	Professeur des Universités	ENS Paris-Saclay	Rapporteur
M. Charbel FARHAT	Full Professor	Stanford University	Président
Mme. Camilla FIORINI	Maîtresse de Conférences	CNAM-M2N	Examinatrice
M. Michel BERGMANN	Directeur de Recherche	INRIA/IMB	Directeur
M. Tommaso TADDEI	Chargé de Recherche	INRIA/IMB	Co-directeur

Membres invités :

M. Jean-Philippe ARGAUD	Ingénieur-Chercheur	EDF R&D	Co-encadrant/Invité
M. Guilhem FERTE	Ingénieur-Chercheur	EDF R&D	Co-encadrant/Invité

Réduction de modèles en mécanique non-linéaire quasi-statique pour l'estimation de l'état par recalage en assimilation de données: application aux enceintes de confinement

Résumé : Dans le domaine de la gestion du parc nucléaire, Electricité de France (EDF) s'efforce d'assurer une compréhension exhaustive de l'état mécanique des enceintes de confinement de ses centrales. Une attention particulière est portée à l'évaluation des taux de fuite à travers les enceintes de confinement à double paroi. Pour atteindre cet objectif, d'importants travaux de recherche ont été entrepris, visant à développer des modèles thermo-hydro-mécaniques (THM) de haute robustesse, spécialement conçus pour la modélisation du vieillissement de ces imposantes structures en béton précontraint. Une étape cruciale consiste à coupler ces modèles à des méthodologies d'optimisation avancées, notamment l'assimilation de données. L'objectif des ingénieurs est d'incorporer les données existantes du parc nucléaire dans ces modèles numériques afin d'obtenir les paramètres physiques optimaux. Cette approche permet de générer des simulations numériques qui reflètent au mieux la réalité observée. Cependant, de tels algorithmes peuvent nécessiter des calculs répétés qui peuvent rendre son coût prohibitif. Dans cette optique, cette thèse vise à développer des méthodes de réduction de modèles pour les problèmes de mécanique non linéaire avec variables internes, ainsi qu'à concevoir des algorithmes permettant de coupler ces méthodologies d'assimilation de données aux modèles réduits, en vue d'accélérer le temps de résolution tout en préservant une qualité d'approximation adéquate. En se basant sur les équations caractéristiques de ces problèmes et sur le contexte industriel, spécifiquement le traitement des problèmes de mécanique non-linéaire quasi-statique avec variables internes dans le code de qualité industrielle `code_aster`, nous détaillons la mise en place d'un modèle réduit par projection utilisant la Proper Orthogonal Decomposition (POD). Cette construction d'une approximation linéaire est ensuite combinée à un processus d'hyper-réduction à travers la méthode ECSW. Dans un second temps, nous avons étendu cette méthodologie à un cas industriel spécifique, à savoir une section courante de l'enceinte de confinement d'une centrale nucléaire. Ce cas implique un matériau nécessitant une modélisation multiple (non linéaire tridimensionnelle pour le béton et linéaire unidimensionnel pour l'acier) avec un comportement mécanique thermo-hydro activé. De plus, nous avons élaboré des algorithmes basés sur les régions de confiance pour aborder des problèmes d'assimilation de données variationnelle en utilisant des modèles réduits. Nous proposons une validation sur des cas en élasticité et des premiers pas sur des cas en mécanique non-linéaire avec `code_aster`. Enfin, nous proposons des méthodes visant à accélérer les processus itératifs mettant en jeu les modèles réduits que nous avons conçu. Cela inclut le développement d'un processus d'hyper-réduction incrémental ou encore une approche bi-fidélité pour le sampling de l'espace paramétrique.
Mots-clés : Réduction de modèles, Assimilation de données, Vieillissement du béton, Mécanique non-linéaire, Optimisation

---

Unité de recherche
UMR 5251 Université, 33400 Talence, France.

Reduced order models in quasi-static nonlinear mechanics for state estimation by calibration through data assimilation: application to containment buildings

Abstract: In the field of nuclear power plant management, Electricité de France (EDF) strives to ensure a comprehensive understanding of the mechanical state of the nuclear containment buildings (NCBs). Special emphasis is placed on evaluating leakage rates through double-walled NCBs. To achieve this objective, major research work has been undertaken to develop highly robust thermo-hydro-mechanical (THM) models, specially designed to model the ageing of these large pre-stressed concrete structures. A pivotal phase involves the integration of these models with advanced optimization methodologies, particularly data assimilation. The engineers aim to incorporate existing data from the nuclear fleet into these numerical models to obtain optimal physical parameters. This approach enables the generation of numerical simulations that reflect observed reality as closely as possible. However, such algorithms can require repeated calculations, which can make them prohibitively expensive. With this in mind, this thesis aims to develop model reduction methods for nonlinear mechanics problems with internal variables, as well as to design algorithms for coupling these data assimilation methodologies to reduced models, with a view to accelerating solution time while preserving adequate approximation quality. Drawing upon the characteristic equations governing these issues and considering the industrial framework, specifically the treatment of quasi-static non-linear mechanics problems with internal variables using the industrial-grade code `code_aster`, we elaborate on the implementation of a projection-based reduced model employing Proper Orthogonal Decomposition (POD). This construction of a linear approximation is then combined with a hyper-reduction process using the ECSW method. In a second step, we have extended this methodology to a specific industrial case, namely a standard section of a NCB. This case involves a multi-modeling material (three-dimensional nonlinear for concrete and one-dimensional linear for steel) featuring thermo-hydro-activated mechanical behavior. Moreover, algorithms based on trust-regions have been devised to address variational data assimilation problems relying on reduced order models. We propose validation on elasticity cases and first steps on nonlinear mechanics cases with `code_aster`. Finally, we introduce methods to expedite iterative processes involving the designed reduced models. These methods encompass the development of an incremental hyper-reduction process and a bi-fidelity approach for sampling the parametric space.

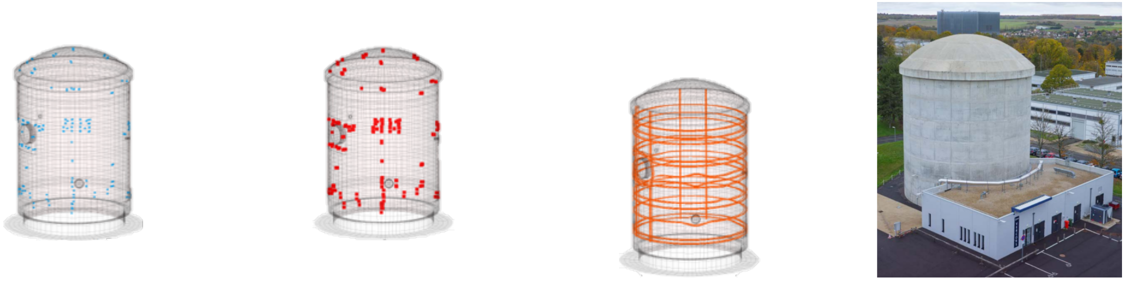
Mots-clés : Model order reduction, Data assimilation, Concrete ageing, Nonlinear mechanics, Optimisation

---

Unité de recherche
UMR 5251 Université, 33400 Talence, France.

Résumé détaillé

En tant que protagoniste majeur dans le secteur nucléaire en France, Electricité de France (EDF) assume un rôle central dans la production et la distribution d'énergie nucléaire. Cette responsabilité englobe la gestion quotidienne des centrales, le maintien des protocoles de sécurité, et l'amélioration globale de l'efficacité opérationnelle. Alors que les premières centrales nucléaires françaises, mises en service à la fin des années 1970, approchent de leur durée de vie opérationnelle initiale de 40 ans, EDF s'attelle à prolonger cette échéance à 60 ans. Le vieillissement des installations se profile ainsi comme un axe de recherche essentiel pour les départements d'ingénierie et de recherche et développement (R&D) de l'entreprise. Un aspect particulièrement critique concerne les enceintes de confinement, qui abritent les cuves des réacteurs. Ces structures, composées de béton, sont sujettes à des problèmes liés au vieillissement, en particulier pour les enceintes dotées d'une double paroi. En effet, les enceintes de confinement à double paroi présentent une possibilité de fuite d'air à travers le béton. Ceci est dû à la porosité de ce matériau. Actuellement, sur les 56 réacteurs nucléaires toujours en exploitation en France, 24 d'entre eux possèdent une enceinte de confinement à double paroi. Pour évaluer et maintenir la capacité opérationnelle continue, des tests de pressurisation décennaux sont effectués sur les enceintes de confinement, les soumettant à des pressions allant jusqu'à 5.2 bars. Un critère crucial dans ce processus est le taux de fuite d'air autorisé à travers les parois, qui ne doit pas dépasser le seuil établi par l'Autorité de Sûreté Nucléaire (ASN) - soit 1.125% de la masse d'air sec par jour. La structure en béton, renforcée et précontrainte, est dimensionnée avec minutie pour résister à des charges accidentelles. Ceci inclut notamment des pressions allant jusqu'à 5.2 bars à l'intérieur de l'enceinte. L'intégrité d'étanchéité des bâtiments de confinement à double paroi dépend de l'état des fissures dans le béton, un phénomène naturel dû à la présence d'armatures. Certaines de ces fissures se produisent pendant la coulée de parties massives ou singulières du bâtiment de confinement, principalement en raison des contraintes imposées pendant la phase de maturation du béton. Durant la phase de précontrainte, ces fissures naturelles sont refermées. Cependant, au fil du temps, le béton subit des contractions dues aux effets de la précontrainte (fluage) et au séchage (retrait), diminuant la précontrainte et réduisant l'action compressive des câbles sur les fissures. De plus, les câbles de précontrainte subissent des déformations dépendantes du temps (relaxation), contribuant à la perte de précontrainte. Cette complexité souligne les défis et les enjeux liés à la prolongation de la durée de vie opérationnelle des centrales nucléaires. Des efforts de recherche sont menés afin de disposer d'une meilleure compréhension des phénomènes mis en jeu, que cela soit par le biais d'essais expérimentaux (voir Figure 1), ou en développant des jumeaux numériques.



(a) Capteurs de déformations. (b) Capteurs de température. (c) Fibres optiques. (d) Maquette VeRCoRs

Figure 1: Visualisation de la maquette VeRCoRs ” *VErification Réaliste du COnfinement des RéacteurS* au EDF Lab sur le site Les Renardières, et des capteurs utilisés. Cette maquette constitue est une reproduction à l'échelle 1/3 d'une enceinte de confinement à double paroi (source: EDF document interne ©).

Cette thèse est motivée par les simulations numériques nécessaires pour le calage de paramètres pour l'étude d'étanchéité des enceintes de confinement à double paroi. La combinaison de données et de modèles numériques, que ce soit pour quantifier les incertitudes (inférence bayésienne) ou l'assimilation de données, relève du domaine de l'optimisation. Les approches numériques de résolution de cette catégorie de problème posent des défis liés à l'efficacité computationnelle, notamment dans des scénarios impliquant de nombreux appels à des codes de calcul. Dans cette thèse, notre principal objectif est de relever les défis liés à une classe particulière de problème, le recalage par assimilation de données. La résolution de ce type de problème nécessite de nombreuses évaluations de modèles, et les coûts computationnels associés à de telles procédures peuvent rapidement devenir prohibitifs lorsqu'un appel au modèle numérique est coûteux. Dans le cadre présenté ici, les ingénieurs se reposent sur des codes de calcul conçus pour simuler des problèmes de mécanique des structures non-linéaire. En particulier, pour la simulation de larges structures en béton précontraint, les ingénieur d'EDF se reposent notamment sur le code de simulation par éléments finis (EF) pour la mécanique des structures `code_aster` [EDF24]. Ce code de qualité industrielle permet aux ingénieurs de définir des simulations numériques de référence, que nous appelons simulations haute-fidélité (HF). Ces dernières sont couplées aux données afin d'identifier les valeurs des paramètres physiques du modèle $\mu \in \mathcal{P}$.

La réduction de modèles regroupe un ensemble d'algorithmes visant à réduire considérablement le coût marginal lié à chaque calcul en exploitant les connaissances acquises lors de simulations HF antérieures. La réduction de modèles paramétriques (pMOR) désigne un ensemble de techniques dédiées à la création d'un modèle de substitution moins coûteux, appelé modèle d'ordre réduit (ROM). L'objectif principal est d'approcher les solutions paramétriques, appartenant à la variété des solutions à l'équation aux dérivées partielles (EDP) paramétrique étudiée, en s'appuyant sur une connaissance de solutions HF pré-calculées. A l'aide de ces dernières, ces approches visent à construire un nouveau modèle numérique dont l'évaluation est moins coûteuse, tout en disposant d'un contrôle sur l'erreur entre les solutions HF et réduites. La méthode des bases réduites est une instance particulière de ces approches où la solution est obtenue par projection du problème HF sur un espace de plus petite dimension. L'approche est profondément enracinée dans le paradigme hors-ligne/en ligne, impliquant une phase hors-ligne (également appelée étape d'entraînement) et une phase en ligne (évaluation de la ROM). Pendant la phase hors-ligne, un ensemble représentatif de solutions HF (appelé *snapshots*) est calculé pour une plage prédéterminée de valeurs de paramètres. Ces solutions servent de base pour construire un espace dit réduit, qui constitue un sous-espace de faible dimension qui capture les caractéristiques essentielles de l'espace des solutions. La phase hors-ligne est intensive en calculs mais n'est effectuée qu'une seule fois. Des techniques telles que la *proper orthogonal decomposition* (POD [BHL93]) sont souvent utilisées pour identifier les modes dominants de variabilité et extraire les informations les plus pertinentes des solutions HF. La phase en ligne implique la résolution d'un problème d'approximation considérablement de plus faible dimension dans l'espace réduit. Cette approche facilite l'évaluation rapide de solutions pour divers paramètres, offrant une réduction substantielle des coûts computationnels par rapport à la résolution du problème d'ordre complet. La phase en ligne est considérée comme réussie lorsqu'elle se caractérise par une efficacité en termes de coût CPU. Historiquement, le paradigme en ligne/hors-ligne trouve son origine dans les structures inhérentes à certaines EDP, en particulier celles présentant des décompositions paramétriquement affines. Cette caractéristique structurelle facilite la décomposition du résidu en une somme de fonctions définies par le produit de coefficients dépendant du vecteur de paramètres et de fonctions indépendantes des paramètres. Cette disposition permet le calcul préalable de nombreux termes pendant la phase hors-ligne, en particulier ceux indépendants des paramètres, permettant ainsi une évaluation rapide de la réponse du système pour des valeurs individuelles de paramètres. Cependant, cette formulation n'est applicable qu'à un ensemble limité de problèmes physiques. Dans les cas où une décomposition efficace du résidu est impossible, l'utilisation d'un espace de faible dimension ne confère pas un avantage computationnel substantiel. Étant donné que l'opérateur est non-linéaire, la complexité computationnelle de l'assemblage de l'opérateur (jacobien et résidus) évolue avec la dimension du modèle HF. Pour surmonter ce défi, des méthodes appelées hyper-réduction sont mises en œuvre pour permettre une évaluation rentable du modèle réduit. Ces approches atténuent efficacement les coûts CPU associés à l'évaluation des résidus non-linéaires. L'ensemble des approches d'hyper-réduction peuvent être classées en deux groupes principaux selon leur philosophie sous-jacente. La première classe vise à construire une décomposition paramétriquement affine pour approcher la forme résiduelle, alignée sur les problèmes de décomposition affine où la méthode RB démontre une efficacité de calcul. Elle inclut par exemple la méthode d'interpolation empirique (EIM [BMNP04]). La seconde classe opère au niveau du maillage afin de diminuer sa taille et de fait, les coûts d'assemblage. De telles techniques incluent la méthode d'hyper-réduction a priori [Ryc05], les approches de quadrature empirique [YP19], la méthode *energy-conserving sampling and weighting* (ECSW [FACC14]) et la méthode *empirical curbature* [HCF17].

Comme mentionné précédemment, nous cherchons à concevoir des approches ROM pour des problèmes d'optimisation dans un cadre industriel, en particulier pour des études de recalage par assimilation de données. L'assimilation de données (AD) est une technique mathématique visant à estimer les variables du modèle μ en combinant des informations antérieures avec des observations (par exemple, des données expérimentales), tout en tenant compte des incertitudes associées. Plus précisément, le recalage du modèle vise à estimer les paramètres du modèle mathématique $\mu \in \mathcal{P}$ à partir d'une sortie d'observation $\mathbf{y} \in \mathbb{R}^{n_{\text{obs}}}$, où $\mathcal{P}$ est la région des paramètres et n_{obs} est le nombre d'observations. Lorsqu'il s'agit de problèmes de mécanique des solides, ces observations peuvent contenir n'importe quelle quantité physique dérivée du champ de déplacement (déplacements, déformations dans les problèmes statiques, vitesses ou accélérations dans les problèmes dynamiques). Le défi des méthodes d'AD réside dans la détermination de la meilleure façon de fusionner des sources d'informations hétérogènes (données et modèles numériques). Les méthodes disponibles dans la littérature se divisent en deux grandes classes [BC02][B$^+$14] : les approches basées sur le filtrage [Kal60], qui reposent sur la manipulation et la mise à jour de matrices de covariance en même temps que l'état; et les approches basées sur le contrôle optimal, connues sous le nom d'approches variationnelles, où l'on suppose que les statistiques des erreurs sont connues et le problème est formulé pour résoudre un problème de minimisation donné. Dans ce travail, nous nous concentrons sur la deuxième classe de méthodes de résolution. Cette dernière repose essentiellement sur la formulation d'une fonctionnelle de coût mesurant l'écart entre les trajectoires possibles du modèle et des observations. En utilisant un opérateur d'observation H^{hf} lié au modèle numérique et un vecteur d'observations physiques $\mathbf{y}$, les paramètres optimaux sont recherchés en tant que minima de la fonction de coût mesurant l'écart entre les mesures et les prédictions du modèle numérique:

$$\mu_{\text{a}} = \arg\min_{\mu \in \mathcal{P}} \text{dist}_{\text{o}} \left(H^{\text{hf}}(\mu),\ \mathbf{y} \right) \tag{1}$$

où $\text{dist}_{\text{o}}(\cdot, \cdot)$ est une distance sur les vecteurs d'observation (expérimentaux et numériques). Afin de résoudre un tel problème, une approche d'AD remarquable est la méthode d'assimilation de données variationnelle tridimensionnelle (3D VAR [Lor86][LDT86][Tal97]). Dans ce cadre, la fonction de coût mesure non seulement l'écart entre les prédictions du modèle et les observations, mais tient également compte des connaissances *a priori* sur les paramètres grâce à un vecteur de paramètres μ_{b}, appelé l'ébauche. D'une part, ce vecteur représente la connaissance *a priori* détenue par un expert et peut être perçu comme un moyen de contourner la non-unicité des solutions en fournissant une connaissance *a priori*. D'autre part, ce processus s'aligne avec une régularisation généralisée de Tikhonov, abordant efficacement la nature mal posée du problème en incorporant des contraintes supplémentaires basées sur les connaissances de l'expert. La régularisation facilite un processus d'AD plus stable, permettant une combinaison judicieuse d'informations dérivées du modèle et de données observées pour affiner les prédictions. Nous nous intéressons donc à la minimisation d'une fonctionnelle:

$$J^{\text{hf}}(\mu) = \frac{1}{2} \left\| \mu - \mu_{\text{b}} \right\|^2_{\mathbf{B}^{-1}} + \frac{1}{2} \left\| \mathbf{y} - H^{\text{hf}}(\mu) \right\|^2_{\mathbf{C}_{\mathbf{R}}^{-1}} \tag{2}$$

où $\mathbf{B} \in \mathbb{R}^{p \times p}$ (resp. $\mathbf{C}_{\mathbf{R}} \in \mathbb{R}^{n_{\text{obs}} \times n_{\text{obs}}}$) est la matrice de covariance de l'ébauche (resp. associée aux observations). Dans le cadre établi de l'optimisation, la technique d'optimisation de région de confiance (TR [CGT00]) émerge comme une méthode de résolution propice à l'usage de ROM. En effet, ces approches reposent sur l'utilisation de modèles d'approximation définis localement à chaque itération du processus d'optimisation, tout en réduisant l'espace des solutions admissibles. Cette dernière condition semble assez intuitive: puisque nous traitons d'un modèle d'approximation local, nous devrions résoudre le problème dans une région où ce modèle d'approximation est suffisamment proche du modèle HF. À travers la méthode TR, nous construisons séquentiellement une série de modèles réduits, générant ainsi un ensemble correspondant de fonctions de coût J_k^{r} (approximation de l'Equation 2) définies en appelant ces modèles d'approximation. De manière itérative, les modèles subissent des modifications jusqu'à ce qu'une estimation fiable du paramètre soit obtenue. Tout l'intérêt de ces approches est d'utiliser des modèles d'approximation dont le coût d'évaluation est plus raisonnable qu'un appel au solveur HF. Une stratégie peut donc consister à utiliser les ROM comme modèles d'approximation pour les approches de régions de confiance.

Les objectifs de cette thèse s'articulent autour de trois axes majeurs. Tout d'abord, notre attention est dirigée vers la formulation de modèles réduits qui soient non seulement efficaces, mais aussi adaptés aux contraintes fixées par le recours à un code de qualité industrielle (`code_aster`). Plus précisément, notre objectif est de construire un modèle réduit adapté aux EDPs associées à nos applications, c'est-à-dire pour des problèmes quasi-statiques en mécanique des structures non-linéaire. Cela nécessite, entre autres, le développement d'un modèle capable d'évaluer avec précision les états des matériaux, facilitant la reconstruction des champs de déplacement et des états de contrainte associés. De plus, une attention méticuleuse doit être portée dans la création des algorithmes et des méthodologies afin de tenir compte des contraintes imposées par l'architecture du code de calcul industriel. La stratégie globale doit fournir une approche capable de traiter un cas d'ingénierie réel: un maillage tridimensionnel représentant une section courante d'enceinte de confinement, incorporant la modélisation du béton précontraint. Un bref examen du cadre théorique imposé par l'opération au sein de ce code est justifié à ce stade. Nous considérons la variable spatiale x dans le domaine Lipschitz $\Omega \subset \mathbb{R}^d$ ($d = 2$ ou 3) et la variable temporelle $t \in [0, t_{\mathrm{f}}]$. Nous introduisons un vecteur de paramètres μ, qui appartient au compact $\mathcal{P} \subset \mathbb{R}^p$, où p est la taille du vecteur de paramètres. Le vecteur μ contient des vecteurs du modèle mécanique utilisé. En effet, dans cette thèse, nous nous limitons aux scénarios où le vecteur de paramètres ne comprend pas de paramètres géométriques. Nous notons u la variable primale du problème (champ de déplacements dans notre cas) et nous introduisons l'espace de Hilbert $(\mathcal{X},\ ||.||_{\mathcal{X}})$ défini sur Ω auquel u appartient. Nous adoptons la notation u_μ afin de souligner la dépendance paramétrique. Bien plus, le cadre établi par les intégrateurs en temps employés dans le code industriel de référence nous oriente vers l'utilisation d'intégrateurs à un pas. D'un point de vue physique, cela signifie que la connaissance de l'état actuel de notre système est dérivée uniquement de l'état calculé précédemment et ignore essentiellement toute information provenant des états précédents. Nous introduisons la liste des pas de temps $\{t^{(k)}\}_{=0}^{K}$, définie comme suit $0 = t^{(0)} \leq \ldots \leq t^{(K)} = t_{\mathrm{f}}$, et nous discrétisons le problème de la manière suivante:

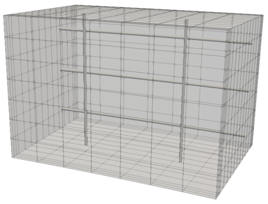
$$u_\mu^{(k)} = u_\mu^{(k-1)} + \Delta u_\mu^{(k)} \quad \text{and} \quad t_\mu^{(k)} = t_\mu^{(k-1)} + \Delta t_\mu^{(k)}, \quad \forall k \in \{1, \ldots, K\} \tag{3}$$

Etant donné $\mu \in \mathcal{P}$, nous cherchons à obtenir la trajectoire temporelle $\mathbb{U}_\mu = \{u_\mu^{(k)}\}_{=0}^{K}$, telle que $\mu \in \mathcal{P}$:

$$\left\{ \begin{array}{rcl} \mathcal{R}_\mu^{(k)} \left(u_\mu^{(k)}, u_\mu^{(k-1)}, v \right) & = & 0, \\ u_\mu^0 & = & \overline{u}_\mu^0,\ \forall v \in \mathcal{X}, \end{array} \right. \tag{4}$$

où $\overline{u}_\mu^0$ désigne la condition initiale et où $\mathcal{R}_\mu^{(k)} \ :\ \mathcal{X} \times \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ la forme résiduelle paramétrique au k-ième pas de temps. Nous étudions une approximation en dimension finie de cette classe de problèmes, en considérant un sous-espace $\mathcal{X}^{\mathrm{hf}} \subset \mathcal{X}$ de dimension finie $\mathcal{N}$. Dans l'ensemble des équations, l'indice ou l'exposant "hf" désigne la discrétisation HF.

Le cas du béton précontraint, modélisé par un couplage entre du béton et de l'acier, entre dans le cadre de ce formalisme. Plus précisément, le problème cible comprend un aspect multi-modélisation où le domaine de simulation Ω peut être décomposé en deux sous-domaines: un pour le béton et l'autre pour l'acier. D'une part, le béton est représenté par un modèle non-linéaire (modèle rhéologique de fluage),



(a) Visualisation du maillage mécanique.

N_e	N_e^{1d}	N_e^{2d}	N_e^{3d}	$\mathcal{N}$	$\mathcal{N}_\mathrm{c}$	$\mathcal{N}_\mathrm{s}$
1532	784	693	55	4076	3911	165

(b) Informations liées au maillage.

Figure 2: Visualisation du maillage mécanique (cf. Figure 2a) et information sur le maillage mécanique (number d'éléments du maillage, que cela soit total ou pour les parties uni- et tri-dimensionnels, Figure 2b).

tandis que les câbles de précontrainte sont décrits par une modélisation unidimensionnelle du comportement élastique linéaire. Des liaisons cinématiques sont effectuées afin de relier les nœuds du béton et les nœuds de l'acier: d'un point de vue théorique, un point dans l'acier et son point coïncident dans le béton sont supposés avoir le même déplacement. Le maillage cible définissant la simulation HF utilisée en pratique pour du recalage par les ingénieurs est présenté dans la Figure 2. Le problème étudié se place dans une étude de couplage faible thermo-hydro-mécanique, où l'état mécanique du matériau est calculé à partir de connaissances de variables auxiliaires précalculées, dont la température et la concentration en eau dans le béton. Dans le cadre de notre étude, cette donnée est à prendre en considération dans la non-linéarité du modèle mécanique, bien que seule la partie mécanique constitue la solution HF à réduire.

Dans le cadre de l'approche de réduction décrite dans ces travaux, nous adoptons une approximation linéaire en construisant une base POD. Cependant, les problèmes étudiés présentent une forte non-linéarité et une dépendance paramétrique non-affine. L'approximation des résidus non-linéaires ne permet donc pas une décomposition efficace hors-ligne/en ligne. Afin de surmonter ce goulet d'étranglement, nous optons pour une approche d'hyper-réduction de quadrature empirique (EQ), de type ECSW [FACC14], qui utilise une repondération des contributions élémentaires. Cette méthode échantillonne un sous-ensemble d'éléments de maillage sur l'ensemble du domaine de calcul pour réduire les coûts d'assemblage lors de l'appel au modèle réduit. Cette approche repose sur la résolution d'un problème de moindres carrés à poids positifs par une approche de type *active-set* [LH95] afin de disposer d'une loi de repondération la plus creuse possible $\boldsymbol{\rho}^{\mathrm{eq}} \in \mathbb{R}^{N_\mathrm{e}}$:

$$\boldsymbol{\rho}^{\mathrm{eq}} = \mathrm{EQ} - \mathrm{solve}\left(\mathbf{G},\ \delta\right) \tag{5}$$

où $\mathbf{G} \in \mathbb{R}^{M \times N_\mathrm{e}}$ est une matrice construite de manière appropriée et δ une tolérance. Grâce à cette approche, un résidu empirique est généré et utilisé lors de la procédure d'assemblage lors de l'appel du solveur ROM, $\forall v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$

$$\begin{aligned}
\mathcal{R}_\mu^{\mathrm{hf},(k)}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \mathbf{v}\right) &= \sum_{q=1}^{N_e} \mathcal{R}_{\mu,q}^{\mathrm{hf},(k)}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right) \\
&\approx \sum_{q=1}^{N_e} \left(\boldsymbol{\rho}^{\mathrm{eq}}\right)_q \mathcal{R}_{\mu,q}^{\mathrm{hf},(k)}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right) \\
&:= \mathcal{R}_\mu^{\mathrm{eq},(k)}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \mathbf{v}\right)
\end{aligned}$$

où $\mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$ est l'espace EF des fonctions vérifiant les conditions aux limites du problème, $\mathcal{R}_\mu^{\mathrm{hf},(k)}$ (resp. $\mathcal{R}_\mu^{\mathrm{eq},(k)}$) la forme HF (resp. réduite) obtenue après discrétisation EF de $\mathcal{R}_\mu^{(k)}$ et $\mathbf{E}_q^{\mathrm{no}}$ est un opérateur de restriction élémentaire pour l'élément q pour les inconnues nodales.

Néanmoins, dans le cadre de la mécanique du solide, la compréhension de l'état mécanique du matériau requiert la connaissance du champ de contraintes sur le maillage HF. Cependant, ces champs sont déterminés par intégration les équations constitutives aux points de quadrature. Du fait de l'adoption d'un maillage réduit, les variables internes ne sont disponibles qu'aux niveaux des éléments échantillonnés dans le maillage. Par conséquent, le champ de contraintes est uniquement connu au niveau du maillage réduit. La reconstruction du champ de contraintes (ou d'efforts généralisés) sur l'ensemble du maillage HF est ensuite réalisée par le biais d'une procédure Gappy-POD [ES95].

Afin d'établir une ROM robuste sur un ensemble de paramètres, nous adoptons une approche gloutonne POD-Greedy [HO08] pour construire le modèle de substitution. Cette méthode itérative vise à enrichir le modèle réduit, comprenant la base et le maillage réduit, en identifiant à chaque itération la solution HF qui est la moins bien approximée par la ROM. La solution la moins précise est évaluée en explorant un ensemble de tests Θ_{train}, défini comme une approximation discrète de l'espace des paramètres $\mathcal{P}$. Nous évaluons les erreurs d'approximation (différence entre la solution HF et la solution réduite) sur l'ensemble des tests afin d'identifier le paramètre pour lequel cette erreur est maximale. Ce paramètre est ensuite utilisé pour améliorer la qualité d'approximation de la ROM. L'extension de cette méthodologie au cas paramétrique nécessite des adaptations dans deux aspects de l'algorithme: la construction de la base et le calcul de l'EQ. Pour la construction de la base, plusieurs possibilités existent: une première approche consiste à réaliser une nouvelle POD sur l'ensemble des snapshots HF calculés; une seconde approche implique une méthode incrémentale, connue dans la littérature sous le nom de H-POD [Haa17]. Cette dernière présente l'avantage de fournir une base hiérarchique obtenue en concaténant la base précédente avec celle obtenue à partir de nouveaux snapshots.

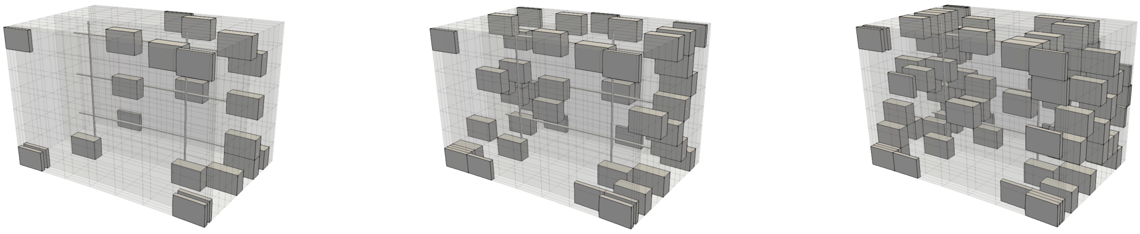
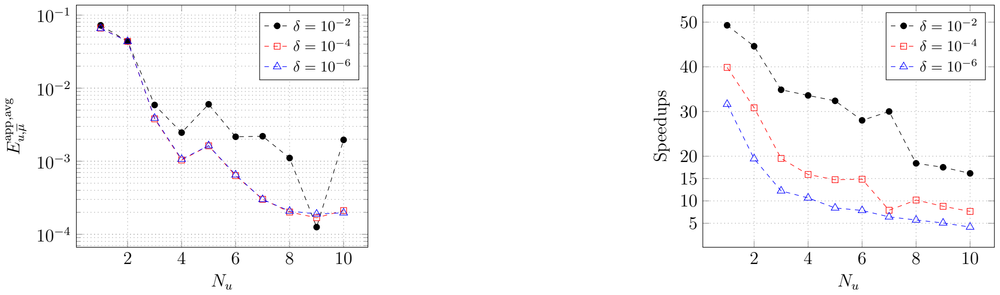

(a) $\delta = 10^{-2}$ (b) $\delta = 10^{-4}$ (c) $\delta = 10^{-6}$

Figure 3: Maillages réduits de la section standard obtenus pour la solution d'un problème de reproduction utilisant $N_u = 5$ modes de déplacement et pour plusieurs paramètres d'hyper-réduction.

L'approche développée permet de disposer d'un sampling effiace pour le maillage réduit et est compatible avec les outils développés au sein d'EDF R&D (voir maillages hyper-réduits de section courante pour un cas non-paramétrique sur la Figure 3). Ceci permet d'accélérer grandement le temps de calcul, même pour un maillage grossier, tout en disposant d'une qualité d'approximation suffisante sur les champs de déplacements. Typiquement, pour un cas non-paramétrique, c'est-à-dire en reproduisant une simulation HF avec la ROM pour un même paramètre, des erreurs relatives moyennées en temps ($E_{u,\overline{\mu}}^{\text{app,avg}}$) de l'ordre du pourcent sont atteintes, avec des accélérations supérieures à 10 en temps CPU (voir Figure 4). Ces bons résultats s'étendent au cas paramétrique, et à l'étude de quantités d'intérêts utilisées en pratique pour les études d'ingénierie (perte de précontrainte dans les câbles) ou les problèmes inverses (champs de déformations au niveau de capteurs).

Dans un second temps, nous proposons une approche de recalage par assimilation de données avec



(a) Erreurs d'approximation sur $\mathbf{u}$.

(b) Speedups.

Figure 4: Évolution des erreurs d'approximation moyennées dans le temps sur les déplacements et les speedups en fonction du nombre de modes utilisés (N_u, cf. Figure 4a) et pour plusieurs tolérances d'hyper-réduction (δ, cf. Figure 4b).

l'utilisation de ROMs afin d'accélérer le processus de recalage des paramètres tout en maintenant la qualité des résultats obtenus. Nous cherchons à éviter délibérément le cadre classique hors-ligne/en ligne prédominant dans de tels contextes de résolution de problèmes. En effet, l'espace des paramètres peut être de grande dimension ($p \gg 1$), rendant la construction d'un modèle réduit *a priori*, trop gourmande en ressources. De plus, les modèles numériques HF sur lesquels reposent ces études subissent des modifications continues, car la modélisation physique du processus est en soi un sujet de recherche. Par conséquent, une ROM construite une fois peut perdre sa validité en peu de temps, suite à des modifications apportées au modèle HF. À la lumière de ces observations, nous avons adopté une approche fondée sur une construction adaptative de la ROM, en développant une approche par région de confiance. Le modèle réduit est construit à la volée en s'adaptant après chaque résolution d'un sous-problème d'optimisation. En effet, l'utilisation d'une approche de région de confiance implique la création d'une suite de modèles d'approximation. Nous résolvons une suite de sous-problèmes d'optimisation, associés à des fonctions de coût J_k appropriées et à une région de confiance $\mathcal{R}_k$ donnée. En conséquence, nous avons une suite de candidats successifs μ_k (centre des régions de confiance) qui convergeront vers une solution, que nous considérons comme la solution à notre problème d'optimisation globale (voir la schématisation de l'algorithme sur la Figure 5). Nous avons développé une approche couplant TR et ROM pour nos problèmes, en nous fondant sur des travaux ultérieurs [Zah16]. Ces approches ont été dans un premier temps validé sur des cas d'élasticité linéaire et non-linéaire, pour différents types d'opérateurs d'observations. Ensuite, des premiers essais numériques ont été menés pour son extension sur `code_aster`.

Cette méthode a été validée numériquement à travers des expériences jumelles, impliquant la création d'un problème avec des données synthétiques pour ajuster un modèle numérique pré-défini. Nous désignons par données synthétiques des informations qui ne proviennent pas d'expériences réelles, mais qui sont extraites d'appels au code de calcul HF. La validation initiale a été effectuée en utilisant le logiciel de simulation `Fenics` [ABH$^+$15] pour des lois de comportement en élasticité linéaire et non-linéaire, tels que le recalage sur un modèle hyper-élastique pour un carré troué soumis à une force volumique (voir Figure 6a). Des opérateurs d'observation locaux ont été testés sur ce cas, en prenant comme observations les déplacements aux points d'intersection entre l'inclusion et les axes (horizontaux et verticaux). Ces expériences ont confirmé la pertinence de la procédure pour notre classe de problèmes, avec des paramètres recalés correspondant aux attentes théoriques, une décroissance des erreurs relatives (voir Figure 6b et 6c). Bien plus, le nombre d'itérations est raisonnable par rapport à des cas HF, où l'algorithme converge en quelques dizaines d'itérations. Comme explicité plus haut, des tests préliminaires ont ensuite été lancés pour étendre cette approche à une implémentation dans le code industriel d'intérêt (`code_aster`).

Enfin, le dernier objectif vise à accélérer les processus de construction de modèles réduits (approche gloutonne) ou d'optimisation (processus itératif) en s'appuyant sur ces modèles réduits. Malgré leur

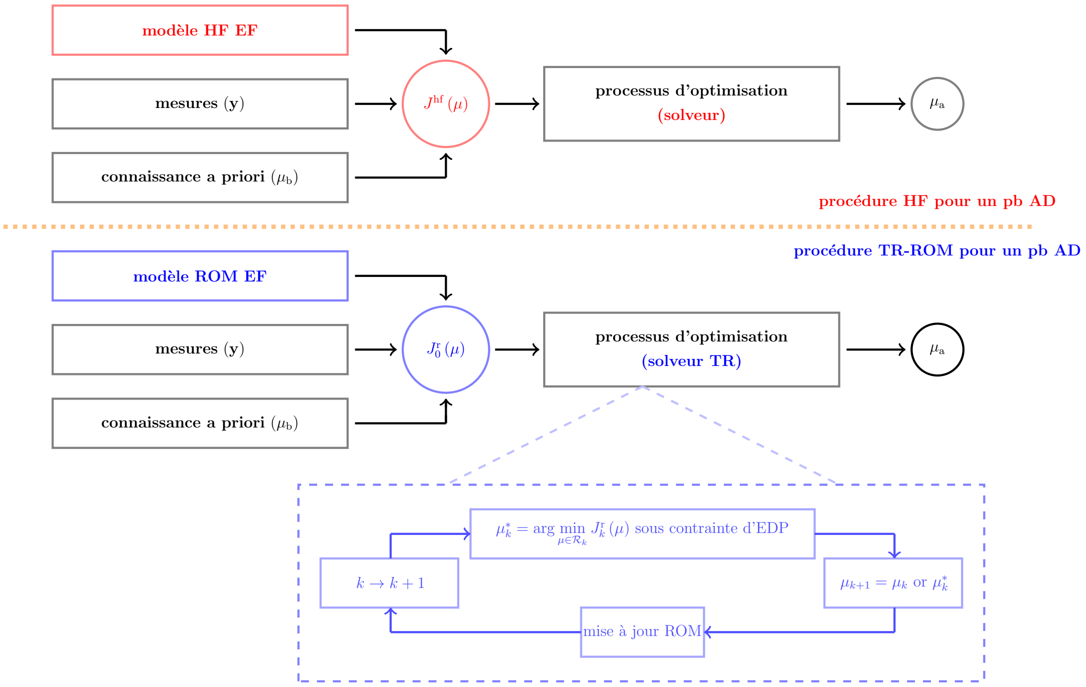
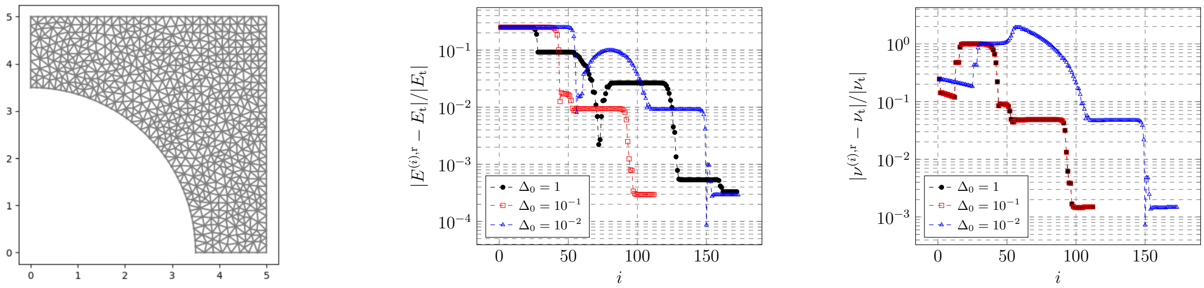


Figure 5: Comparaison d'une approche de recalage par assimilation de données pour le cas HF et pour le cas d'une approche par régions de confiance avec ROM. Cette dernière approche implique une oucle interne au sein du processus d'optimisation afin de mettre à jour la ROM au cours des itérations.



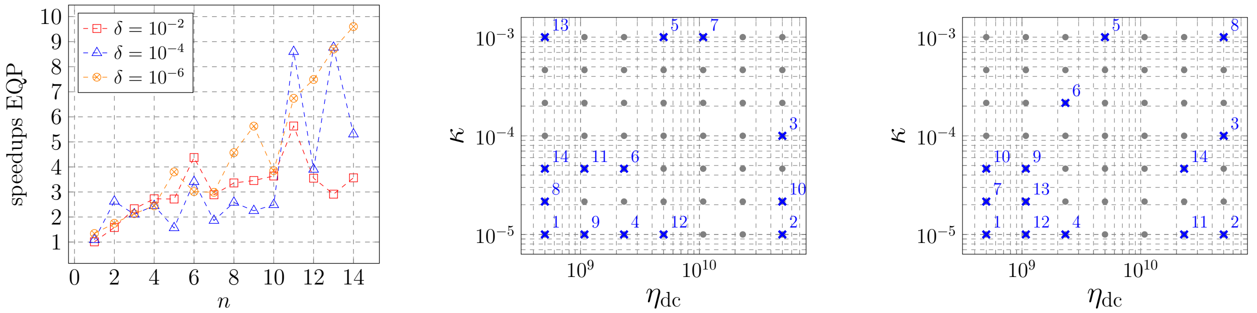
(a) Maillage HF utilisé dans un cas hyper-élastique (`Fenics`)

(b) Erreurs E pour un opérateur d'observation local.

(c) Erreurs ν pour un opérateur d'observation local.

Figure 6: Géométrie utilisée pour un cas hyper-élastique (élasticité non-linéaire) d'un carré troué, avec un opérateur d'observation local, et erreurs relatives sur les paramètres entre les itérés $\mu^{(i),r}$ et le vrai paramètre μ_t.

efficacité, les méthodes mentionnées précédemment peuvent entraîner des coûts computationnels substantiels, qui peuvent entraver l'application industrielle. Nous proposons donc plusieurs approches afin de réduire certains coûts hors-ligne mis en jeu dans ces procédure. D'une part, les processus impliquant la procédure EQ reposent sur des problèmes d'optimisation au cours de la phase hors-ligne (cf. Equation (5)). Le coût de calcul pour résoudre ces problèmes augmente avec le nombre de lignes (représentant les *snapshots* et les modes: essentiellement la taille des espaces d'approximation et d'entraînement) et de colonnes (indiquant la taille du maillage). Nous proposons donc une approche incrémentale pour accélérer ces processus.



(a) Speedups entre approche incrémentale et standard.

(b) Sampling paramétrique sur un maillage grossier.

(c) Sampling paramétrique sur un maillage raffiné.

Figure 7: Résultats des méthodologies d'accélération appliquée à un maillage raffiné de section courante d'enceinte: speedups de la méthode d'hyper-réduction incrémentale, et visualisation des sampling de paramètres pour les maillages grossiers et raffinés pour $\mu = [\eta_{\mathrm{dc}}, \kappa] \in \mathbb{R}^2$.

La méthodologie proposée se fonde sur l'hypothèse de bases incrémentales (H-POD) pour lesquelles les dictionnaires successifs se construisent aussi de façon incrémentale. L'ensemble des indices actifs pour une itération précédente est alors pris comme point de départ pour le nouveau problème d'optimisation. Une telle approche permet typiquement d'accélérer grandement les calculs (speedups pouvant aller jusqu'à 10) dans le cas d'un maillage raffiné de section courante d'enceinte (voir Figure 7a). D'autre part, nous étudions la faisabilité d'une approche bi-fidélité pour échantillonner l'espace paramétrique. Plus précisément, nous étudions des maillages de différentes résolutions (maillage grossier et raffiné) pour générer un échantillonnage de l'espace paramétrique par processus glouton. Pour les expériences numériques menées, nous présentons ici par exemple le cas industriel avec un paramètre à deux dimensions $\mu = [\eta_{\mathrm{dc}}, \kappa] \in \mathbb{R}^2$. Ici, η_{dc} représente la viscosité de fluage de dessiccation, et κ est le paramètre de consolidation pour le fluage propre. Les *samplings* effectués explorent globalement des zones similaires de l'espace paramétrique. Bien plus, la qualité des ROMs obtenues est similaire. Il peut donc être intéressant d'utiliser une telle stratégie afin de minimiser les coûts d'appel à la procédure HF lors du processus glouton.

Ces travaux ont élaboré et mis en œuvre une approche de réduction de qualité élevée pour l'approximation des problèmes étudiés, en particulier dans le contexte industriel. L'approche par région de confiance avec modèles réduits semble être une solution prometteuse pour aborder efficacement les problèmes de recalage par assimilation en mécanique des structures. En se basant sur des aspects incrémentaux ou sur plusieurs niveaux de fidélité, ces implémentations peuvent être rendues robustes et applicables dans un contexte industriel. Ceci inclut notamment une réflexion sur la minimisation des coûts offline. En conclusion, les perspectives futures sont vastes, que ce soit pour étendre l'application des modèles réduits à des problèmes plus vastes et coûteux, ou pour aborder les problèmes d'assimilation de données dans des cas réels.

Acknowledgments

Je voudrais remercier en premier lieu tout encadrement de thèse pour ces trois années de grande qualité Dr. Michel Bergmann, Dr. Tommaso Taddei mes research supervisors du côté de l'Inria, et Dr. Jean-Philippe Argaud et Dr. Guilhem Ferté, mes encadrants à EDF R&D. Je remercie Michel pour sa hauteur de vue, ses conseils de grande qualité et son calme dans l'avancement de la thèse qui m'ont permis de mener ma recherche sereinement dans un bon cadre de travail. Je tiens à remercier Tommaso pour sa disponibilité de tout instant, les discussions de recherche extrêmement intéressantes, mais aussi pour les critiques constructives, et les conseils afin de m'améliorer pour grandir en tant que chercheur. J'ai beaucoup appris sur la rédaction, la manière de formuler des idées à ses côtés et je le remercie grandement. Je tiens à remercier Guilhem pour m'avoir donné l'opportunité de réaliser cette thèse, de m'avoir toujours aidé quand j'en avais besoin, malgré un emploi du temps très chargé! Je le remercie pour son aide et son soutien pour gérer la thèse tout au long des trois années. Je tiens à remercier grandement Jean-Philippe, pour une disponibilité sans faille, toutes les semaines, pour les réponses même en vacances! Je tiens à le remercier pour ces longues discussions passionantes et débats qui duraient parfois des heures. Ce fut très important pour moi d'être accompagné et écouté tout au long de ce parcours de thèse. Ses conseils furent d'une grande aide notamment pour apprendre à présenter des résultats de manière efficace. Je tiens à remercier Sylvie Michelle-Ponnelle et Angélique Ponçot, qui bien qu'elles n'étaient pas officiellement dans l'encadrement, ont pris beaucoup de temps, chacune, pour m'aider et m'accompagner à différents moments de la thèse et pour me donner des retours très constructifs sur mes travaux.

Je tiens à remercie Prof. Ludovic Chamoin, et Prof. Virginie Ehrlacher pour avoir accepté d'être rapporteur de mes travaux de recherche et Prof. Charbel Farhat et Dr. Camilla Fiorini pour avoir accepté d'être dans mon jury. Je suis honoré d'avoir la chance de présenter mes travaux à de tels chercheurs, dont j'admire le travail.

Je tiens à remercier mes collègues d'EDF R&D qui m'ont accompagné tout au long de ces trois années et avec qui j'ai partagé d'excellents moments. Mickael et sa bonne humeur quotidienne, mes camarades d'O2BC Thomas et Nicolas pour les débats politiques et les bonnes bières, mes "co-bureau" pleins de bonne humeur, que ce soit Ting puis Zouhair, mon fidèle soldat depuis les Ponts Anthony, mon acolyte d'espiègleries à EDF Donatien, et mon sanglier préféré Matthieu. Je tiens à remercier tous les doctorants/jeunes embauchés avec qui j'ai passé de bons moments, Amine, Hao, Ana-Clara, Eliass, Antoine, Taryk, Amalio, Thibault, Youri, Idrissa. Vous avez tous rendu cette expérience quotidienne extrêmement agréable, et m'avez d'ailleurs convaincu de rester parmi vous, c'est dire! Je ne peux pas citer tout le monde et j'espère que ceux que je n'ai pas cités ne se vexeront pas.

Je tiens aussi à remercier les membres de l'équipe Memphis à Bordeaux, qui m'ont toujours bien accueilli quand je suis venu, et avec qui j'ai passé de bons moments, notamment Beatrice, Karl, Nishant, Giulia et Giuliano qui m'ont accompagné une bonne partie de la thèse. Je tiens à remercier Angelo et Anne-Laure, qui, chacun à leur manière et avec leurs responsabilités, m'ont aidé lorsque j'en avais besoin, et ont rendu mes séjours à Bordeaux extrêmement agréables.

Je tiens à remercier tous mes camarades dans ma vie quotidienne en dehors de la thèse, qui ont pu m'accompagner pendant ces trois ans. Mes coéquipiers de mon équipe de foot, mes amis depuis les Ponts, la classe préparatoire ou les lyonnais depuis le lycée. Merci à tous ceux qui m'accompagnent quotidiennement depuis plus ou moins longtemps dans la vie, j'ai la chance de vous avoir. Une mention spéciale à tous ceux que je vois régulièrement sur Paris: Adrien, Thomas, le duo Régis-Nadège, Baptiste, Marius, Paul, Romane, Anto (même si tu nous as quitté pour la vie londonnienne), et à tous les fidèles soldats de mon club de foot.

Je tiens à remercier ma famille, mes deux parents, pour m'avoir amené jusqu'ici. J'espère que vous serez fier de moi, comme je suis fier de votre soutien pendant toutes ces années! Merci pour m'avoir donné la chance d'arriver jusqu'ici, par mon éducation. Merci aussi à celui qui partage ma vie au quotidien Xav, et qui doit me supporter tous les jours, ce qui n'est pas chose facile. Merci pour son soutien, et sa présence à mes côtés pendant toutes ces années!

Contents

List of Figures

List of Tables

Acronyms

ASN Nuclear Safety Authority. 3

BCs Boundary Conditions. 17
BFGS Broyden-Fletcher-Goldfarb-Shanno. 108

CIWAP Civil Work Assessment Project. 4
CPU Central Processing Unit. 7

DA Data Assimilation. 8
DOF Degree of Freedom. 5

EDF Electricité de France. 3
EQ Empirical Quadrature. 34

FE Finite element. 6
FOM Full-Order Model. 34

HF High-Fidelity. 5

ICs Initial Conditions. 19

KKT Karush-Kuhn-Tucker. 42

MOR Model Order Reduction. 6

NCB Nuclear Containment Building. 2
NNLS Non-Negative Least Squares. 36

PDE Partial Differential equation. 10
pMOR Parametric Model Order Reduction. 6
POD Proper Orthogonal Decomposition. 7

PWR Pressurized Water Reactor. 2

QoI Quantity of Interest. 66

RB Reduced Basis. 6

ROM Reduced-Order Model. 6

RSV Representative Structural Volume. 55

SVD Singular Value Decomposition. 29

TR Trust-region. 8

VeRCoRs VErification Réaliste du COnfinement des RéacteurS. 3

Chapter 1

Introduction

1.1 Industrial context and motivations

1.1.1 A historical perspective on nuclear energy in France

The quest for energy has been a pivotal driver of societal development throughout our contemporary history. Initially, the abundance of energy was ushered in by fossil fuels in the 19th century, subsequently complemented by the advent of nuclear power and renewable energies, catalyzing a transformative evolution in our societies. Energy stands as a fundamental cornerstone for the functionality of our modern economies and the independence of nations or supranational entities, particularly during periods of heightened political or economic instability. It is within this context that the French nuclear industry flourished during the latter half of the 20th century.

Figure 1.1: EDF nuclear fleet in operation in France in 2022. EPR Flamanville 3 under construction. The digit assigned to each location indicates the count of reactors situated at each power station (Source: IRSN ©- `https://www.irsn.fr`).

The oil shocks of the 1970s, acting as pivotal triggers, significantly influenced the trajectory of the French nuclear program. However, the allure of nuclear energy for French engineers predates these economic crises, as evidenced by the initial developments of a civil nuclear program in the 1960s. Indeed, the first generation of reactors (graphite-gas technology) was tested with the construction of nine reactors starting in 1963. Due to techno-economic reasons, the pressurized water reactor (PWR) technology was adopted from 1968 onwards. Armed with pre-established knowledge of the required technologies, substantial investments were made in the French nuclear industry, shaping the nation's electrical grid. Consequently, France established an extensive nuclear power infrastructure during the latter half of the 20th century, incorporating multiple generations of reactors. This strategic move positioned France as a prominent player in the global nuclear landscape, symbolizing not only energy security but also technological prowess. Nowadays, the current fleet of electricity-generating reactors in operation in France consists of a total of 56 PWRs, known as "Generation II," and one European Pressurized Water Reactor (EPR) currently under construction in Flamanville (Manche), referred to as "Generation III" (cf. Figure 1.1).

The development of this extensive nuclear fleet allowed France to significantly reduce its fossil energy imports. Indeed, while primary energy production tripled between 1973 (514 TWh) and 2021, the share of nuclear power rose from 9% to 75%. The development of nuclear power in France is therefore an essential part of the country's energy strategy in order to meet climate commitments or to maintain competitive electricity production costs. A distinctive feature in France is the standardized nature of the fleet: all 56 PWRs utilize the same technology and are technically akin. They are spread across 18 nuclear power plant sites.

1.1.2 Nuclear containment buildings and leakage

A nuclear reactor can be broadly divided into two parts: a "nuclear island" where nuclear fission produces heat, and a "conventional island" where this heat is converted into electrical current. The nuclear island includes crucial components such as the reactor building, housing the reactor itself and the entire pressurized primary circuit, as well as some systems ensuring the reactor's operation and safety. The nuclear containment building of a nuclear power plant is the third and final containment barrier, in case of accident, between the radioelements inside the reactor and the outside world (cf. Figure 1.2). Nevertheless, the design philosophy of nuclear containment buildings (NCBs) in French nuclear plants has evolved progressively since the early 1970s.

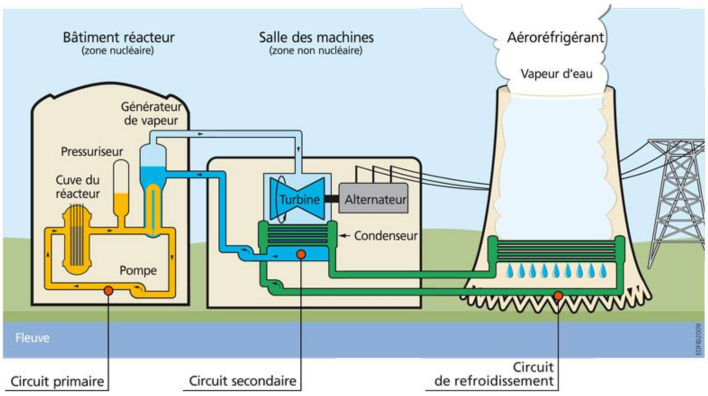


Figure 1.2: Functional principle of a nuclear power plant with air cooler (IRSN ©- `https://www.irsn.fr`).

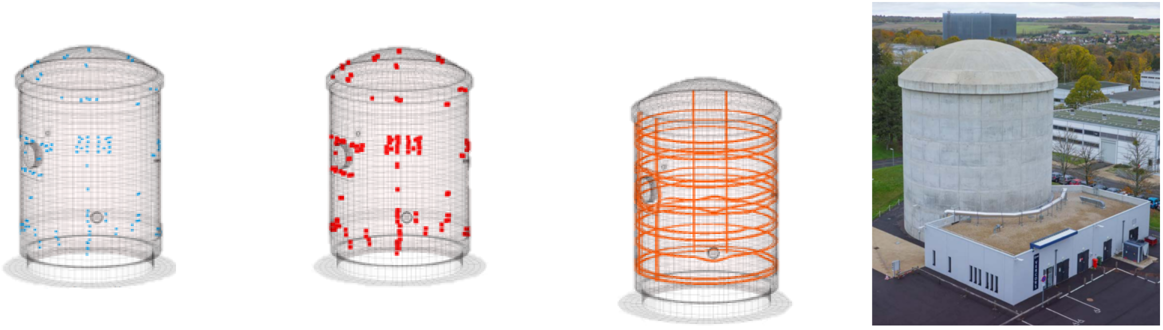
While NCBs initially consisted of a single prestressed concrete wall, with leak-tightness ensured by a steel liner, the ensuing designs featured double concrete walls, comprising an inner prestressed concrete wall without a steel liner and an outer reinforced concrete wall. From a simplified point of view, the outer wall ensures reactor protection from external aggressions like weather or accidental aggressions like aicraft fall, while the inner wall guarantees leaktightness in case of internal incident or accident. Moreover, an active system maintains a constant vacuum between the two walls, allowing potential radioactive elements from internal accidental situations to be retained and filtered rather than released into the environment.

Within the French nuclear industry landscape, Electricité de France (EDF), as the primary operator of nuclear facilities in France, is tasked with the generation and distribution of nuclear energy. Its responsibilities encompass the day-to-day management of nuclear power plants, maintenance of safety protocols, and the overall operational efficiency of the nuclear energy sector. Moreovern EDF adheres to the standards set by the Nuclear Safety Authority (ASN). Initially, the design of the power plants was based on a 40-year operational lifespan. Given that the first plants became operational in the late 1970s, the design assumption is approaching its limit. EDF, as the operator of the French nuclear fleet, has expressed a desire to extend the operational lifespan of the reactors to 60 years, as this was already done in the US. Thus, plant ageing has become a major research theme for EDF's engineering and R&D departments. The containment building, which houses the reactor vessel, is a non-replaceable component of the structure and thus highly prone to ageing-related problems. In the current state of the French nuclear fleet, 24 of the 56 nuclear reactors still in operation feature a so-called double-walled containment. As mentioned above, these NCBs possess a special technical feature which entails a particular risk: the leaking possibility of high pressurized gas in case of accident through the concrete. Since concrete is a porous medium, gases can diffuse through the structure and be released, in part, into the outside environment. The evolution of this phenomenon may be affected by ageing.

The continual operational capability is substantiated through a decennial pressurization test of the containment structures up to 5.2 absolute bars. The quantified air leakage within the inter-envelope space must not surpass the criterion stipulated by the ASN (1.125% of dry air mass per diem). Furthermore, the reinforced and biaxially prestressed concrete structure is meticulously dimensioned to endure inadvertent loads, encompassing 5.2 bars of pressure and a temperature of 150°C within the internal containment. The sealing integrity of double-walled NCBs depends on the state of concrete cracking in the structure, a natural occurrence in reinforced concrete due to the presence of reinforcements. Some of this cracking occurs during the casting of massive or singular parts of the containment building, primarily due to the constraints imposed during the concrete maturation phase. During the prestressing phase, this natural cracking is closed. During the operational phase, it is essential to maintain a sufficient state of prestress continuously to keep these cracks closed. However, concrete contracts over time due to prestress effects (creep) and drying (shrinkage), shortening the prestressed cables and reducing their compressive action on the cracks. The high-strength steel cables, too, undergo time-dependent deformations (relaxation), contributing to prestress loss.

1.1.3 Experimental insights into double-walled nuclear containment leakage phenomenon

As part of the investigations related to these containment structures, the understanding of the phenomenology of delayed deformations in large prestressed concrete structures constitutes a domain of expertise in which EDF has invested significant resources. The resulting research program led to the construction of a 1/3 scale model of the containment structures of the P'4 reactor fleet, named VeRCoRs for " *VErification Réaliste du COnfinement des RéacteurS*" (Realistic Verification of Reactor Containment). 30 metres high and 16 metres in diameter, VeRCoRs is the first research model of this size. The VeRCoRs model, a reduced-scale reproduction of an authentic double-walled containment structure (P'4 platform), was erected at the EDF Lab Les Renardières site. This collaborative initiative involves diverse EDF units with shared objectives and began in August 2013. This testing facility, by virtue of its scale, expedites the ageing process of enclosures within the nuclear park. Its primary objectives encompass the comprehension and modeling of the physical phenomena underlying ageing, their intricate interactions, and their implications on leakage rates. This enhances prognostic capabilities concerning the evolution between tests, its extrapolation to accident scenarios, and its precise localization. Furthermore, the facility substantiates the robustness of double-walled NCBs in the context of accident situations. From the standpoint of engineering disciplines, research endeavors have particularly elucidated that the temporal progression of prestress loss in cables represents a crucial parameter in the investigation of leakage rates. Furthermore, a multitude of research initiatives has been dedicated to these thematic areas. Modeling endeavors have been undertaken to formulate suitable constitutive laws [Koa23] and numerical models for the simulation of such structures. Besides, investigations are led to scrutinize the sensitivities of outcomes to parameters [Bou18] and to quantify uncertainties [Ros22].



(a) Deformation sensors

(b) Temperature sensors

(c) Optical fibers

(d) VeRCoRs mock-up

Figure 1.3: View of the mock-up: sky view of the mock-up in EDF Lab Les Renardières site, and layout of the various sensors within the mock-up (source: EDF internal document ©).

This structure, distinguished by its exceptional size for a research facility, also stands as a beacon of digital innovation (cf. Figure 1.3). Some sensors mirror those found in current reactors but are notably more abundant, while others, such as optical fibers, represent novel additions being meticulously tested. From an experimental perspective, the mock-up helps to assess and check methodologies before they can be deployed throughout the fleet, or to reflect on the measurement tools to be installed during the construction of future power plants. Furthermore, engineers aspire to realize the ambitious concept of a digital twin, seamlessly integrating real data (e.g., derived from the VeRCoRs model) with numerical models to optimize parameters for physical models, and delineate their associated values and uncertainties. This acquired knowledge not only enables the anticipation of the future trajectory of the containment structure but also furnishes engineers with estimates of critical quantities like leakage rates. Once mastered, this knowledge could be extrapolated to apply specific methodologies across the entirety of the nuclear fleet. The VeRCoRs model stands as a noteworthy effort within EDF's R&D department's extensive Civil Work Assessment Project (CIWAP). This comprehensive project encompasses endeavors related to understanding and modeling leakage from double-walled NCBs, with VeRCoRs model being a significant component thereof. The first phases of the project focused on modeling the mechanical behavior of the model. Several tasks are now underway, including the transposition of this work to the oldest NCBs in the fleet, and the design of numerical twins based on the models already established. This thesis falls within the scope of this project supported by EDF R&D.

1.1.4 Digital twin design and challenges for robust and efficient numerical simulation

This thesis is motivated by the numerical simulations involved when tackling the challenge of the leak-tightness of double-walled NCBs. Indeed, combining data and numerical models—whether for quantifying uncertainties (Bayesian inference) or data assimilation— pertains to the domain of optimization problems and poses challenges related to computational efficiency, particularly in extensive many-query scenarios. Within this thesis, our primary focus is on addressing challenges related to data assimilation. Nevertheless, these challenges demand a multitude of model evaluations, and the computational expenses associated with such procedures can swiftly become prohibitive.

Focusing on the analysis of extensive pre-stressed concrete structures, and grappling with the complexities of numerical simulation within civil engineering, engineers heavily depend on computational codes designed for non-linear structural mechanics. Indeed, the numerical models employed in these investigations rely on industrial-grade finite element codes, specifically designed for engineering computations in real-world scenarios. Specifically, for the numerical simulations at stake here, engineers employ the structural mechanics software `code_aster` [EDF24], a robust finite element code developed by EDF R&D for addressing intricate engineering challenges. `code_aster` boasts a rich history dating back to the 1980s, originating from the imperative recognized by EDF to create a sophisticated numerical simulation tool for the nuclear and energy sectors. Its continuous refinement over decades has led to a versatile and widely used tool, known for its proficiency in handling multiphysics simulations, encompassing structural mechanics and heat transfer, and adeptly simulating complex nonlinear mechanical behaviors. Moreover, `code_aster` is an open-source software, fostering collaboration and enabling users to customize and extend its functionalities based on their specific needs.

CODE aster

Figure 1.4: Logo of the high-fidelity industrial-grade finite element code for structural mechanics developped at EDF R&D (`https://code-aster.org`).

Computations for simulations entail a substantial time commitment to achieve numerically accurate solutions for the addressed problems. Indeed, certain applications demand a high number of degrees of freedom (DOFs) to ensure solutions of adequate quality. Besides, the models used are nonlinear, featuring couplings (both physical and geometric) whose resolution (solvers, integration of constitutive equations) can be intricate. More precisely, engineers define reference simulations, which we call high-fidelity (HF) simulations. Regarding our application example, they typically concern a portion of an NCB, chosen to meet criteria set by the engineers. These are simulations whose input data are the physical parameters of nonlinear structural mechanics models. In our scenario, the HF partial differential equation undergoes discretization through the finite-element

(FE) method for structural mechanics.

However, implementing these models within a computational framework that involves numerous iterations is challenging in practice. Indeed, when algorithms repeatedly invoke HF simulations, they may become impractical. Therefore, it may be imperative to develop algorithms aimed at alleviating the computational costs of these numerical procedures in view of overcoming bottlenecks in industrial applications.

1.2 Reduced order models for data-assimilation problems

Within this thesis, all research is framed within the scope of reduced order modeling for data assimilation challenges within an industrial context. More precisely, the work must conform to the numerical algorithms and codes employed within the industrial framework and be suitably adapted to seamlessly integrate into data assimilation procedures. Indeed, the overarching goal is to develop models that are more cost-effective than current ones, tailored to data assimilation challenges. These efforts need to adhere to the specificities of the studied physical models while also navigating the constraints imposed by industrial frameworks. In the following paragraphs, we first offer a general overview of the employed methodology and then delineate the objectives of our thesis work. Subsequently, we briefly introduce the development environment used for these research endeavors.

1.2.1 Methodology overview

Model order reduction (MOR) consists in a broad spectrum of algorithms that aim to drastically reduce the marginal cost linked to individual computations by leveraging insights gained from earlier HF simulations. Parametric model order reduction (pMOR) denotes a set of techniques dedicated to crafting a low-dimensional surrogate model, known as reduced-order model (ROM). The primary goal is to approximate the solution field across a spectrum of parameters, drawing upon insights derived from preceding HF simulations. One of the purposes of these methods is to propose a new numerical model, build upon knowledge frome the HF numerical scheme, and whose evaluation is affordable, while ensuring similar approximation quality. The latter is often evaluated by means of the approximation error, which corresponds to the error between the HF solution, also known as the full-order solution, and the reduced solution. The whole point of the approach is to have control over the approximation errors on the parametric manifold under study. Reduced Basis (RB) method is a model order reduction technique employed in the context of parametrized PDEs. The core idea behind RB method is to construct a low-dimensional subspace, known as the reduced basis space, wherein the solution is sought, and therefore, reducing the computational burden associated with solving the problem for each parameter independently. The reduced space is made of a a small number of global basis functions, build through precomputed solutions (called snapshots) of the HF model. Further details on the RB method can be found in the following non-exhaustive reference list [RHP08][QMN15][HRS$^+$16]. The RB method is deeply rooted in the offline/online paradigm. The latter is a systematic strategy which involves a twofold process: an *offline* phase (also called training stage) and an *online* phase (cf. Figure 1.5). During the offline phase, a representative set of HF solutions is computed for a predetermined range of parameter values. These HF solutions serve as a basis for constructing a reduced basis space, a low-dimensional subspace that captures the essential features of the solution manifold. The offline phase is computationally intensive but is performed only once, independent of the specific parameters during the online phase. The constructed reduced basis space forms the crux of the online phase. Techniques such as Proper Orthogonal Decomposition (POD [BHL93]) are often employed to identify dominant modes of variability and extract the most relevant information from the HF solutions. When faced with a new parameter value, the online phase entails solving a considerably lower-dimensional approximation problem in the reduced space. This approach facilitates the swift assessment of solutions for diverse parameters, offering a substantial reduction in computational expenses compared to addressing the full-order problem.The online phase is deemed successful when marked by efficiency in terms of CPU cost, making it suitable for real-time applications, or optimization studies.

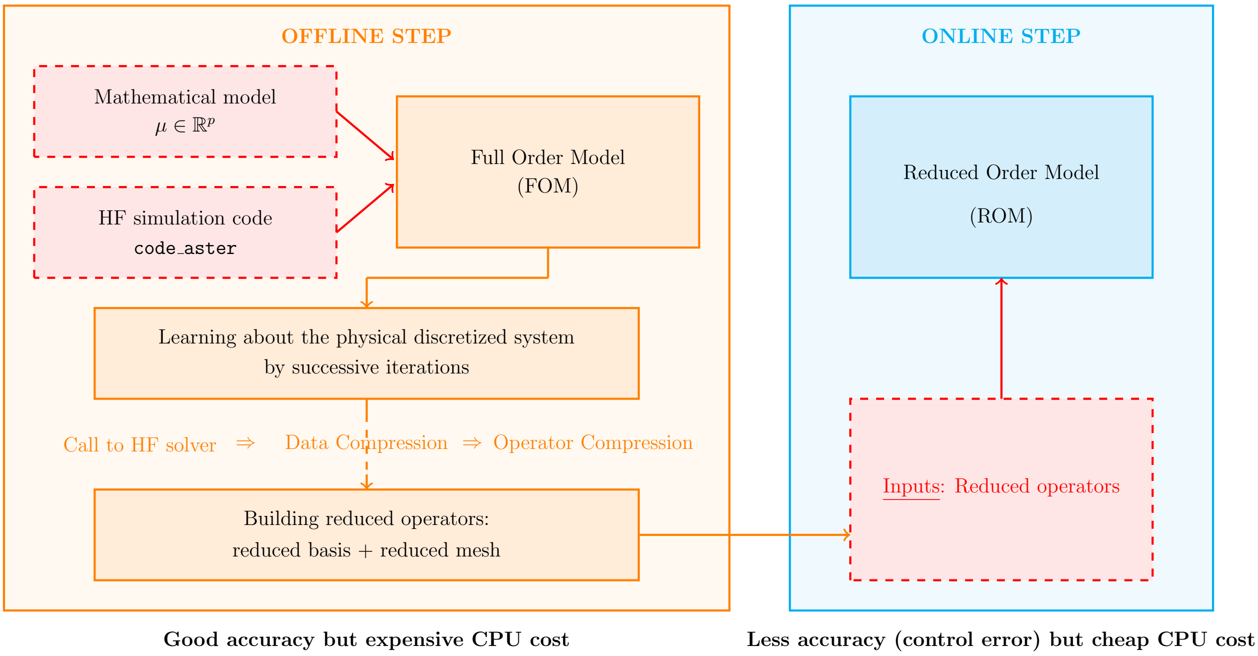


Figure 1.5: Visualization diagram of the online/offline paradigm: separation between an a priori phase of ROM construction, executed only once, with high computational cost, and an online phase of calling with lower CPU cost.

Historically, the online/offline paradigm originated from the inherent structures of some PDEs, particularly those exhibiting parametrically affine decompositions, a characteristic prominently present in elliptic problems. This structural feature facilitates the decomposition of the residual into a sum of functions defined by the product of coefficients dependent on the parameter vector and functions independent of parameters. Such an arrangement allows for the precomputation of numerous terms during the offline phase, specifically those independent of parameters, thereby enabling a rapid evaluation of the system response for individual parameter values. However, this formulation is only applicable to a limited set of physical problems, particularly those of a nonlinear nature. In scenarios where an efficient decomposition of the residual is unattainable, employing a low-dimensional space does not yield a substantial computational advantage. Since the operator is nonlinear, the computational complexity of the operator assembly (jacobian and residuals) scales with the size of the HF model. To overcome this challenge, methods known as hyper-reduction are implemented to enable a cost-effective evaluation of the reduced model. These approaches effectively alleviate the computational expense associated with evaluating nonlinear residuals. Further details on these methods are elucidated in Chapter 2.

As mentioned above, we seek to design ROMs for industrial optimization problems, in particular for data assimilation studies. Data assimilation (DA) is a mathematical technique, whose aim is to seek an estimation of model variables μ by combining prior information with observations (e.g., experimental data) while considering the associated uncertainties. More precisely, model calibration (or updating) aim to estimate parameter of the mathematical model $\mu \in \mathcal{P}$ from an observation output $\mathbf{y} \in \mathbb{R}^{n_{\text{obs}}}$, where $\mathcal{P}$ is the parameter region and n_{obs} the number of observations. When considering solid mechanics problems, those observation may contain any physical quantity derived from the displacement field (displacements, strains in static problems, velocities or accelerations in dynamical problems). The challenge for DA methods lies in determining how best to blend heterogeneous sources of information (data and numerical models). The set of available methods in the literature falls into two broad classes [BC02][B$^+$14]: filtering-based approaches [Kal60], which rely on the handling and updating of covariance matrices at the same time than the state; and optimal control-based approaches, known as variational approaches, where we assume that the errors statistics are known, and the problem is stated so as to solve a given minimisation problem. In this work, we focus on the second class of resolution methods. The latter is essentially founded on the formulation of a deviation function between the model's possible trajectories and the observations. Given an observation operator H^{hf} tied to the numerical model, and a vector of physical observations $\mathbf{y}$, optimal parameters are sought as minima of the cost function measuring the deviation between measurements and numerical model predictions:

$$\mu_{\text{a}} = \arg\min_{\mu \in \mathcal{P}} \text{dist}_{\text{o}}\left(H^{\text{hf}}\left(\mu\right),\ \mathbf{y}\right) \tag{1.1}$$

where $\text{dist}_{\text{o}}\left(\cdot,\cdot\right)$ is a distance on the observation vectors (experimental and numerical). Like all inverse problems, challenges of ill-posedness, following Hadamard's definition [Had02], and ill-conditioning issues are inherent in model updating problems, exacerbated particularly by the presence of measurement noise and model bias. Addressing these challenges requires the consideration of regularization techniques to effectively navigate and solve the associated optimization problem. One prominent DA approach is the Three-Dimensional Variational Data Assimilation (3D VAR) method. Within this framework, the cost function not only measures the misfit between model predictions and observations, but also takes *a priori* knowledge on the paremeters into account thanks to a parameter vector μ_{b}, called the background. The latter embodies the *a priori* knowledge held by an expert, encompassing an understanding of the influence of parameters on solution shapes (and observations) or an awareness of the expected order of magnitude of parameters for the utilized experimental setup. The introduction of the term related to the background is then perceived as a means to circumvent the non-uniqueness of solutions by providing a priori knowledge. This process aligns with a generalized Tikhonov regularization [Tik], effectively addressing the ill-posed nature of the problem by incorporating additional constraints based on expert insights. The regularization facilitates a more stable assimilation process. It relies on a judicious combination of model-derived information and observed data to refine predictions and improve the accuracy of the assimilated results. Thus, the cost function J^{hf} is expressed as a weighted sum of quadratic terms incorporating all the above information (cf. Figure 1.6). This algorithm is a robust method, historically developed in the meteorological community [Lor86][LDT86][Tal97]. More details are given on the exact formulation of this optimization problem in Chapter 5.

Within the established optimization framework, the trust-region (TR [CGT00]) optimization technique emerges as a viable solution. The key idea behind these approaches involves using local approximation models during the optimization iterations, while reducing the set of admissible solutions. This last condition seems fairly intuitive: since we are dealing with a local approximation model, we should solve the problem in a region where this approximation model is sufficiently close

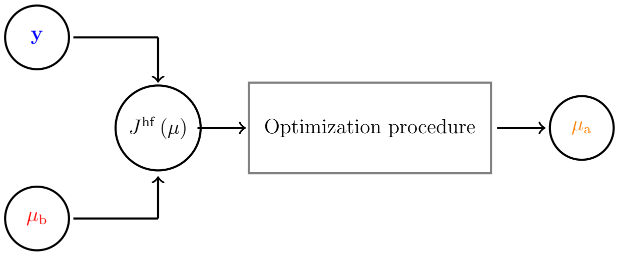


Figure 1.6: General principles of a 3D-VAR data-assimilation problem given an observation vector $\mathbf{y}$ and a background vector μ_{b}.

to the HF model. The approximation models should be less expensive to evaluate in order to gain in terms of CPU cost. Trust-regions approaches offers a formulation that accommodates the use of less computationally demanding approximation models during the optimization iterations. In the context of ROM design, one strategy may be to use ROMs as approximation models for trust-regions approaches for a specific class of optimization problems namely DA problems. Through the TR method, we sequentially construct a series of reduced models, thereby generating a corresponding set of cost functions $\{J_k^{\mathrm{r}}\}$ defined on these approximation models. In an iterative fashion, the models undergo modifications until a reliable estimate of the parameter is achieved. Figure 1.7 elucidates the distinction between the conventional approach, relying on the HF code, and the methodology that integrates TR with ROMs.

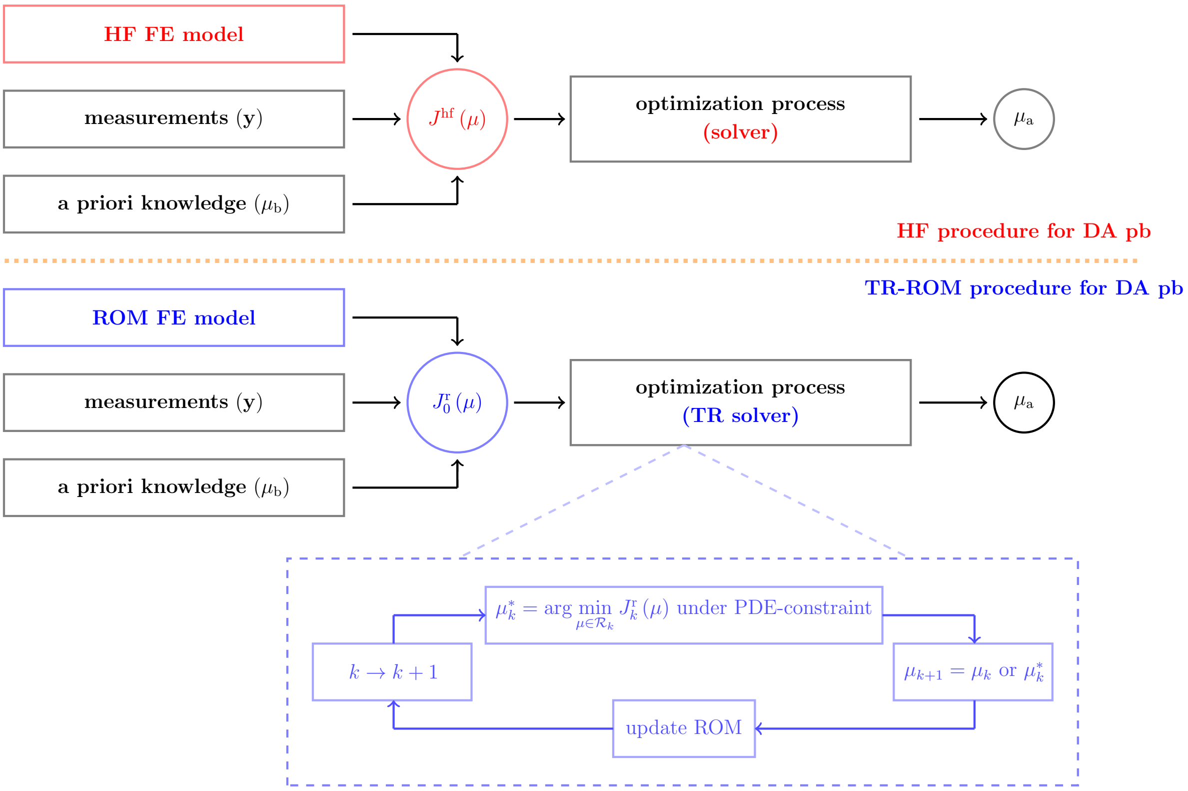


Figure 1.7: Comparison of the data-assimilation procedure under an HF scheme versus the use of trust-region with ROM. The call to the TR solver induces a internal loop in the optimization process with an update of the ROM.

1.2.2 Objective of the work

The objectives of this thesis can be delineated into three primary topics. Initially, our attention will be directed towards formulating ROMs that are not only efficient but also well-suited for the prevailing industrial challenges. Specifically, our goal is to construct a ROM tailored to the pertinent partial differential equations, addressing quasi-static problems in nonlinear structural mechanics. This requires, among other criteria, the development of a model proficient in accurately assessing material states, facilitating the reconstruction of displacement fields and associated stress states. Additionally, our implementations must align with the digital tools employed by engineers, accommodating the constraints imposed by industrial architecture and computational codes. Within the framework of this thesis, all methodologies have been devised within an industrially recognized finite element simulation code extensively used in structural mechanics: `code_aster`. This reality underscores the need for meticulous consideration in crafting the algorithms and methodologies. The overarching strategy should furnish both a systematic approach and a high-quality ROM to address a real-world engineering challenge: a three-dimensional mesh representing a portion of a containment structure, incorporating prestressed concrete modeling. A brief examination of the theoretical framework mandated by the operation within this code is warranted at this juncture. Further insights into modeling intricacies and decision-making processes will be expounded upon in Chapter 3.

We consider the spatial variable x in the Lipschitz domain $\Omega \subset \mathbb{R}^d$ ($d = 2$ or 3) and the time variable $t \in [0, t_\mathrm{f}]$. We introduce a vector of parameters μ, which belongs to the compact parameter region $\mathcal{P} \subset \mathbb{R}^p$, where p is the number of parameters. The μ vector contains physical parameters of the mechanical models used. In this thesis, we limit our consideration to scenarios where the parameter vector does not include geometric parameters. This decision is made to avoid the complexity associated with mesh mapping techniques or more intricate methods required to handle parametric dependencies. Furthermore, the industrial challenges within our framework do not consider variations in geometry. Thus, such a parametric configuration falls outside of the scope of our work. We denote by u the primal variable of the problem and we introduce the Hilbert space $(\mathcal{X}, \|.\|_{\mathcal{X}})$ defined over Ω to which u belongs. We adopt the notation u_μ to highlight parametric dependency. In a static setting, the problem of interest may be stated as such:

$$\text{find } \ u_\mu \in \mathcal{X} \ : \ \mathcal{R}_\mu\,(u_\mu,\ v) = 0, \quad \forall v \in \mathcal{X},\ \mu \in \mathcal{P} \tag{1.2}$$

where $\mathcal{R}_\mu \ : \ \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the parametric residual associated with the partial differential equation (PDE) of interest.

At the core of our investigation lies the temporal evolution of a nuclear containment structure. This particular problem is characterized by its non-stationary nature. More precisely, our focus is on formulations of quasi-static problems, ones that encompass a temporal evolution while neglecting inertia terms. The framework established by the time-integrators employed in the reference industrial-grade code directs us towards the utilization of one-step integrators. From a physical standpoint, this means that the knowledge of the current state of our system is derived solely from the previously computed state and essentially disregards any information from preceding states. We introduce the time grid $\{t^{(k)}\}_{=0}^{K}$, defined as $0 = t^{(0)} \leq \ldots \leq t^{(K)} = t_\mathrm{f}$, and we discretize the problem as outlined below:

$$u_\mu^{(k)} = u_\mu^{(k-1)} + \Delta u_\mu^{(k)} \quad \text{and} \quad t_\mu^{(k)} = t_\mu^{(k-1)} + \Delta t_\mu^{(k)}, \quad \forall k \in \{1, \ldots, K\} \tag{1.3}$$

Given $\mu \in \mathcal{P}$, we seek the sequence $\mathbb{U}_\mu = \{u_\mu^{(k)}\}_{=0}^{K}$, such that, for all $\mu \in \mathcal{P}$:

$$\left\{ \begin{array}{rcl} \mathcal{R}_\mu^{(k)}\left(u_\mu^{(k)}, u_\mu^{(k-1)}, v\right) & = & 0, \\ u_\mu^0 & = & \overline{u}_\mu^0,\ \forall v \in \mathcal{X}, \end{array} \right. \tag{1.4}$$

where $\overline{u}_\mu^0$ denotes the primal initial solution of the problem at stake and where $\mathcal{R}_\mu^{(k)} : \mathcal{X} \times \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the parametric residual at the k timestep. In summary, our focus revolves around quasi-static non-linear problems, which can be high-dimensional. This class of problems encompasses many structural mechanics problems. Besides, it is worth noting that the initially presented static problem (cf. Equation (1.2)) can also be considered as part of this broader class of problems.

We examine a finite-dimensional approximation of this class of problem (cf. Equation (1.4)), by considering a subspace $\mathcal{X}^{\mathrm{hf}} \subset \mathcal{X}$ of finite dimension $\mathcal{N}$. This approximation is referred to as HF discretization, and throughout the following sections of the manuscript, the 'hf' subscript or superscript designates HF discretization. Within this thesis, we employ continuous Lagrange FE basis functions for discretizing the problems at hand, which represents the standard numerical approach employed in structural mechanics codes. Our first objective is to construct a model that is less computationally demanding than this HF problem, while ensuring consistency in the obtained results. The physical models formulated, the discretization methods utilized, and the architecture of these codes constitute input data for our problem, to which we must tailor our approach.

Second, we intend to couple a data assimilation approach with the use of ROMs to speed up the process of parameter recalibration while upholding the quality of the yielded results. Our approach aligns with the following paradigm: we deliberately avoid the classic offline/online framework prevalent in such problem-solving contexts. Indeed, such an approach would involve building a ROM valid over a parameter range before employing it for data assimilation purposes. This choice would entail studying the influence of approximation error on the result of the data assimilation process and proposing solutions to correct the problem coherently. However, this type of process is not entirely suitable for the aforementioned industrial context. Indeed, the parameter space can exhibit a notably high dimension, rendering the construction of a ROM too resource-intensive. Furthermore, the numerical models on which these studies rely undergo continuous modification since the physical modeling of the process is a reseach topic in itself. Consequently, a ROM constructed once may lose its validity within limited time, following modifications to the HF model. In light of those remarks, we have embraced an approach where data assimilation is based on an adaptive construction of the ROM, thereby accelerating the data assimilation process. In this scenario, the ROM is constructed exclusively to address the specific problem at hand, aiming to expedite the process without necessarily being suitable for all studied problems. Although this loss of generality may be acknowledged, this approach contributes to curtailing the offline expenses linked with ROM construction. Moreover, it also fits in neatly with the desired industrial framework, where the modeling of constitutive equations is an ongoing area of research frequently subjected to challenges. The goal is to establish an on-the-fly approach to optimization problems, with a specific focus on addressing data assimilation challenges within a variational framework using ROMs.

Finally, the last objective aims to accelerate the construction processes of ROMs (greedy) or optimization (iterative process) banking on these reduced models. Despite their efficiency, the previously mentioned methods can incur substantial computational costs if applied directly. Hence, our approach revolves around a judicious use of ROMs in an industrial context, considering several aspects, namely reducing some offline costs, which can hinder industrial application.

1.2.3 Development environment

A fundamental aspect of this work lies in the fact that, rather than being exclusively geared toward industrial applications, the advancements were executed within the framework of `code_aster`. This choice was not only a necessity owing to the intricate nature of the models to be simulated but also a strategic decision to formulate a methodology that aligns seamlessly with the data structures and algorithms employed in practical applications. A HF numerical simulation typically involves input files representing meshes and measurements, yielding output files containing numerical results. Development work has therefore been carried out in order to ensure consistent calls with `code_aster` data structures.

As part of a collaborative project (FUI Mordicus) involving various universities and companies, EDF contributed to the development of a Python module for model reduction calculations. The objective was to amalgamate methods, such as data compression, optimization, and hyper-reduction techniques, independent of HF codes. The methodologies presented in this thesis were integrated with the `mordicus` package and `code_aster` to facilitate knowledge capitalization within R&D tools. All algorithms were implemented and incorporated into the `mordicus` Python code, while the hyper-reduction methods were embedded in the core of `code_aster`, complementing the existing projection reduction capabilities within this code.

1.3 Contributions and outline of the thesis

1.3.1 Contributions

The key contributions of this work can be summarized as follows:

- the development of projection-based reduced-order models (ROMs) for parametric quasi-static problems in nonlinear mechanics, and extension to multi-modeling framework for the accurate simulation of standard sections of NCBs,
- the developement and implementation of an POD-greedy adaptive approach within an industrial-grade broadly used FE code `code_aster` and numerical tests on different real-world problems namely: three-dimensional elasto-plastic plate with holes, and a standard section of a double-walled NCB,
- the adaptation of trust-region approaches using ROMs for data-assimilation problems, and the application of this methodologies to nonlinear solid mechanics problems;
- the improvement of the ROM building and the hyper-reduction process for iterative procedure relying ROMs, namely greedy approaches or optimization processes. This encompass incremental hyper-reduction and multi-fidelity strategies for the parameter sampling.

1.3.2 Structure of the thesis

Given the overarching goals of the thesis, our work can be divided in three parts: first, the construction of projection-based ROMs for quasi-static problems in nonlinear solid mechanics; second, the development of a data assimilation algorithm based on the coupling between ROMs and trust region approachs; thirds, acceleration strategies for iterative processes relying on ROMs.

Chapter 2 furnishes an extensive review of reduction methods pertinent to our research context. The methodological decisions inherent in our work and the numerical validation of all introduced approaches are expounded upon in Chapters 3 and 4. We present two main cases: an elasto-plastic case and a representative structural volume of a NCB. The first can be seen as an initial step in the development and validation of the method for a classical nonlinear mechanics problem with a single modeling approach. The second case extends the method to a multi-modeling scenario for a problem dependent on precalculated variables. Indeed, this industrial case involves a kinematic coupling between a three-dimensional model and a one-dimensional model. The rheological model used for concrete (three-dimensional part) is at the core of a weak coupling of multiple physics. Consequently, the material behavior is influenced by previously calculated physical fields. The representative structural volume case is a continuation of previous research efforts on a model developed and studied in various research works [Bou18][Bou16]. Engineers commonly use this test case in practical studies concerning NCBs.

Subsequently, our attention is directed toward the entirety of the work pertaining to data assimilation processes. An overview of existing methods and the theoretical framework for this category of assimilation problems is provided in Chapter 5, followed by an in-depth exposition of the methodology developed in our work and a numerical demonstration of the quality of the optimization process.

To conclude, Chapter 6 proposes numerical strategies for accelerating the iterative processes presented in the other sections.

1.3.3 Publications and communications

Publications and communications related to this thesis are listed hereafter.

- **Publication in peer-reviewed journals**:
    - ◇ [AAB^{+}24b] Eki Agouzal, Jean-philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. International Journal for Numerical Methods in Engineering, 125(4):e7385, 2024.
- **Preprints**:
    - ◇ [AAB^{+}22] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. *arXiv preprint arXiv:2212.14825*, 2022,
    - ◇ [AAB^{+}24a] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, Sylvie Michel-Ponnelle, and Tommaso Taddei. Projection-based model order reduction for prestressed concrete with an application to the standard section of a nuclear containment building. *arXiv preprint arXiv:2401.05098*, 2024,
    - ◇ [AT24] Eki Agouzal and Tommaso Taddei. Accelerated construction of projection-based reduced-order models via incremental approaches. *arXiv preprint arXiv:2401.07108*, 2024.

◇ **Communications in international and national conferences**:

- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2022. *Réduction de modèles de problèmes paramétriques en mécanique non linéaire à l'aide de Code Aster et Mordicus.* 15ème Colloque National en Calcul des Structures (CSMA),
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2022. *Model reduction approach for parametric quasi-static non-linear mechanics problem in an industrial codes.* Congrès d'Analyse Numérique (CANUM),
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2022. *Parametric model order reduction approach for quasi-static non-linear mechanical problems using an industrial code: application to an elasto-plastic material.* MORE,
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2023. *Projection-based model order reduction for large-scale nonlinear problems in structural mechanics using an industrial code.* SIAM CSE,
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2023. *Projection-based model order reduction for multi-modelling problems in nonlinear structural mechanics.* M2P,
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2023. *Projection-based model reduction for creep behavior in prestressed concretes.* YIC,
- ◇ Agouzal E., Argaud J.P., Bergmann M., Ferté G., Taddei T. 2023. *Projection-based model order reduction for a representative structural volume of a nuclear containment building.* MorTECH.

Chapter 2

Model order reduction approaches

2.1 Introduction

This chapter provides a detailed exploration of the model reduction methodology developed and applied in this thesis, establishing essential connections with prior research. It begins by detailing the specific physical problems under scrutiny and then outlining the numerical methods employed to design a suitable ROM for these challenges.

First, Section 2.2 focuses on presenting the specific problems addressed within this work, namely nonlinear quasi-static problems in structural mechanics. The description of weak formulations of the mechanical problems and their relationships with previously introduced formulations (cf. Equation (1.2) and (1.4)) are stated. We also explain the discretization methods used, giving the reader a comprehensive understanding of the scope of the problems investigated in this thesis. More specifically, the ultimate objective of this work resides in the numerical simulation of prestressed concrete within the framework of an industrial code. In order to elucidate the procedure, we opt to address two issues, herein referred to as the single-modeling approach and the multi-modeling approach. The former pertains to the application of a ROM in the numerical simulation of a material exhibiting nonlinear behavior with the presence of internal variables; the latter category of issues involves more intricate physical models that account for the coupling between multiple materials and mechanical behaviors activated by physical phenomena (that means, with weak coupling to other physical phenomena). The first category can be viewed as a simplified version of the second. As a reminder, the latter is the motivation behind our work.

Moving ahead, our focus shifts to a comprehensive exploration of the methodology in use. We will contextualize this examination within the broader landscape of current research initiatives. This involves critical decisions about choosing the reduced model method, exploring the approximation space, and addressing operator nonlinearity (see Section 2.3). Finally, we detail the notion of *a posteriori* error indicators and the choice made in a single-modeling case (see Section 2.5).

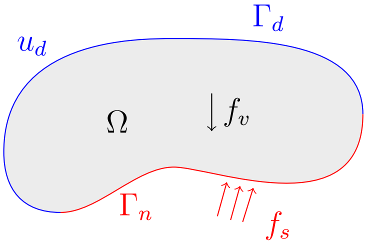
2.2 Nonlinear quasi-static problems in structural mechanics

2.2.1 Problem formulation

In our earlier discussion, we introduced the computational domain Ω and the time variable $t \in [0, t_\mathrm{f}]$. Our focus lies on nonlinear mechanical problems involving internal variables, specifically two classes of problems whose PDEs align with the aforementioned framework. First, we investigate the scenario of a single material, representing a mechanical system with a unique constitutive equation. In the upcoming text, we denote this scenario as a single-modeling formulation. Subsequently, we introduce the more intricate case of a multi-modeling structure, characterized by kinematic coupling between computational domains and the possibility of weak multi-physics coupling. This latter case corresponds to the mechanical model employed for prestressed concrete in NBCs, and we will provide a detailed exploration of this model later in our discussion (see Chapter 3, more specifically Section 3.3.3).

As previously stated, our attention is directed towards problems within the realm of structural mechanics, where the primary unknown of the system is the displacement field u. We concentrate on nonlinear small-displacement small-strain mechanical problems with internal variables. Within our context, the constitutive equations are inherently nonlinear, and the system implicitly depends on the history of displacements through a differential equation that incorporates nonlinear behaviors such as elastoplasticity or viscoplasticity. Our approach adopts the general framework of materials with internal variables, where these variables serve as secondary parameters in a mechanical problem. They help capture the internal state of the material and contribute to describing its history-dependent behavior. The study of internal variables is particularly relevant in the context of irreversible thermodynamics and constitutive modeling, where it allows for a more accurate representation of material responses beyond the traditional elastic regime.

Single-modeling formulation



As mentioned above, we shall begin our discussion with the case of single-modeling. To be more precise, we deal with the case of a three-dimensional mechanical system that can be modeled using a two or three-dimensional approaches ($d = 2$ or 3). We adopt the framework of continuum mechanics for small deformations. Within this framework, a comprehensive understanding of the mechanical state involves the knowledge of the displacement field (u_μ), the stress field (expressed through the Cauchy tensor σ_μ), and the internal variables (γ_μ). In this work, we address only quasi-static formulations, which means that we omit the inertial term from the equilibrium equations. Time evolution is described by the system of ordinary differential equations in each point in Ω:

$$\left\{\begin{array}{rcl} -\nabla\cdot\sigma_\mu & = & f_v \\ \sigma_\mu & = & \mathcal{F}^\sigma_\mu\left(\nabla_s u_\mu,\ \gamma_\mu\right) \\ \dot{\gamma}_\mu & = & \mathcal{F}^\gamma_\mu\left(\sigma_\mu,\gamma_\mu\right) \end{array}\right. \qquad + \text{ Boundary Conditions (BCs)} \tag{2.1}$$

where the nonlinear operator $\mathcal{F}^\sigma_\mu$ stands for the constitutive equation that maps the state of stresses in the material from the knowledge of deformations (∇_s is the symmetric part of the gradient, $\nabla_s\cdot = \frac{1}{2}(\nabla\cdot + \nabla^\top\cdot)$) and internal variables, while the nonlinear operator $\mathcal{F}^\gamma_\mu$ denotes an equation of evolution of internal variables within the material. The first equation in the system below describes the equilibrium of our system. The evolution equation for internal variables encapsulates the dynamics of these variables, serving as carriers of information regarding the material's history. These variables play a crucial role in characterizing how the material responds to external influences over time. At last, the constitutive equation provides us with the stress state of the material. In our formalism, and within the context considered, the stress state, identified by the Cauchy stress tensor σ_μ, is determined by both the deformation of the system ($\nabla_s u_\mu$) and the material's history (internal variables γ_μ). This equation sheds light on how the material structurally responds to both instantaneous deformation and its historical interactions, offering a comprehensive understanding of its mechanical behavior.

As previously indicated, the time-discretization schemes introduced in this context are specifically tailored for one-step integrators (cf. Equation (1.3)). In other words, our approach involves a backward Euler discretization scheme for the evolution equation such that the quasi-static discretization of the system leads to the following formulation:

$$\left\{\begin{array}{rcll} -\nabla\cdot\sigma^{(k)}_\mu & = & f^{(k)}_v & \text{on} \quad \Omega \\ \gamma^{(k)}_\mu & = & \gamma^{(k-1)}_\mu + \Delta t^{(k)}\mathcal{F}^\gamma_\mu\left(\sigma^{(k)}_\mu,\gamma^{(k)}_\mu\right) & \text{on} \quad \Omega \\ \sigma^{(k)}_\mu & = & \mathcal{F}^\sigma_\mu\left(\nabla_s u^{(k)}_\mu,\ \gamma^{(k)}_\mu\right) & \text{on} \quad \Omega \\ & + & \text{BCs} & \end{array}\right. \tag{2.2}$$

Theoretically, stresses can be considered as internal variables. Building upon this formulation, we choose for convenience to restate the problem by displaying only the stress variable in our formulation, as follows:

$$\left\{\begin{array}{rcll} -\nabla\cdot\sigma^{(k)}_\mu & = & f^{(k)}_v & \text{on} \quad \Omega \\ \sigma^{(k)}_\mu & = & \mathcal{F}^{(k)}_\mu\left(u^{(k)}_\mu,\ u^{(k-1)}_\mu,\sigma^{(k-1)}_\mu\right) & \text{on} \quad \Omega \\ & + & \text{BCs} & \end{array}\right. \tag{2.3}$$

where $\mathcal{F}^{(k)}_\mu\left(.,.\right)$ is an appropriate nonlinear operator. In this framework, internal variables are seen as an inner part of the operator $\mathcal{F}^{(k)}_\mu$. As the internal variables take into account the history of the material, we can either write the residual as an explicitly time-dependent form, or incorporate the internal variables into the formulation to explicitly denote the dependency. We emphasize that our methodology is appropriate for problems of the form (2.2), although we further define it for problems of the form (2.3).

We now explore the incorporation of boundary conditions (BCs) into our formulations. As previously mentioned, our methodology aligns with the computational framework dictated by the software used for numerical tests. In the context of the numerical tools employed, we implement Dirichlet-type conditions to account for linear combinations of state variables. This involves a linear form c linked to kinematic BCs, assuming the displacement field lies within the kernel of this form. Such a selection allows for the representation of arbitrary linear relations on the displacement field. Beyond handling homogeneous Dirichlet BCs, it supports various scenarios, such as the uniform translation of an unknown amplitude of a subpart or any other arbitrary linear relationship between the DOFs of Ω that captures kinematic links between subsystems of the overall mechanical system. In our work, we consider the general case that encompasses both non-homogeneous Neumann and homogeneous Dirichlet BCs for appropriate linear combinations of state variables:

$$\left\{\begin{array}{rclcc} \sigma_\mu^{(k)} \cdot n &=& f_s^{(k)} & \text{on} & \Gamma_n \\ c(u_\mu^{(k)}) &=& 0 & \text{on} & \Omega \end{array}\right. \tag{2.4}$$

where n is the outward normal to the boundary Γ_n, and $f_v^{(k)}$ (resp.$f_s^{(k)}$) is the volumic (resp. surfacic) force applied to the system, and c the previously mentionned linear form. The variational form of the equilibrium equation given by Equation (2.1) reduces to the following residual expression:

$$\mathcal{R}_\mu^\sigma\left(\sigma_\mu^{(k)},\ v\right) = \int_\Omega \sigma_\mu^{(k)} : \varepsilon(v)\ dx - \int_\Omega f_v v\ dx - \int_{\Gamma_n} f_s v\ ds, \qquad \forall v \in \mathcal{X} \tag{2.5}$$

This yields a conventional formulation of the residual form articulated in continuum mechanics, known as the principle of virtual work. Finally, the variational problem investigated in this contribution can be summarized as follows $\forall k \in \{1, ..., K\}$:

$$\text{Find } u_\mu^{(k)} \in \mathcal{X}_{\text{bc}} \text{ s.t. },\quad \left\{\begin{array}{ccc} \mathcal{R}_\mu\left(u_\mu^{(k)},\ u_\mu^{(k-1)},\ \sigma_\mu^{(k-1)},\ v\right) = 0 \quad , & & \forall v \in \mathcal{X}_{\text{bc}} \\ \sigma_\mu^{(k)} = \mathcal{F}_\mu^{(k)}\left(u_\mu^{(k)},\ u_\mu^{(k-1)}, \sigma_\mu^{(k-1)}\right) & \text{on} & \Omega \\ \sigma_\mu^{(k)} \cdot n = f_s^{(k)} & \text{on} & \Gamma_n \end{array}\right. \tag{2.6}$$

where $\mathcal{X}_{\text{bc}} := \{v \in \mathcal{X} :\ c\,(v) = 0,\ \text{on } \Omega\}$. The test space matches the functions belonging to the kernel of the linear form of the boundary conditions. To provide a more general formalism close to the previous notations, we detail the relationship between the two residual formulations as follows, $\forall v \in \mathcal{X}$:

$$\mathcal{R}_\mu\left(u_\mu^{(k)},\ u_\mu^{(k-1)},\ \sigma_\mu^{(k-1)},\ v\right) = \mathcal{R}_\mu^\sigma\left(\mathcal{F}_\mu^{(k)}\left(u_\mu^{(k)},\ u_\mu^{(k-1)}, \sigma_\mu^{(k-1)}\right),\ v\right). \tag{2.7}$$

A quick rephrasing reveals that the formulation of this problem aligns with the framework established by Equation 1.4, $\forall v \in \mathcal{X}$:

$$\mathcal{R}_\mu^{(k)}\left(u_\mu^{(k)},\ u_\mu^{(k-1)}, v\right) = \mathcal{R}_\mu\left(u_\mu^{(k)},\ u_\mu^{(k-1)},\ \sigma_\mu^{(k-1)},\ v\right),$$

The sole distinction between the two residuals lies in the incorporation of time dependency. This implies that each residual is expressed in a distinct manner. In contrast, the alternative formulation opts for a singular residual, albeit necessitating knowledge of the mechanical state from the preceding iteration. However, both formulations are equivalent.

Multi-modeling formulation

We now consider the modeling of large prestressed concrete structures. Prestressed concrete is a construction material that incorporates compressive stresses induced by tensioned high-strength steel strands or tendons before the concrete is poured. This process enhances the performance and durability of the concrete structure. For the mentioned structures, the initial step involves pouring concrete around sheaths, allowing it to initiate the drying process. Subsequently, cables are inserted into these ducts and prestressed to meet civil engineering standards. Finally, cement is poured into the ducts. Once the cement has hardened, the tendons are anchored, creating a state of compression within the concrete. This compression counteracts the tensile stresses that the structure may experience during its service life. In order to model such a material, the developed mechanical model is built on a coupling between a three-dimensional model (modeling the concrete) and a one-dimensional model (modeling the prestressing steel cables). We assume that the domain Ω can be split into a three-dimensional domain Ω^{c}, and a one-dimensional domain Ω^{s}. The latter can be decomposed in $n_{\mathcal{C}}$ cables $\Omega^{\mathrm{s}} = \{\mathcal{C}_i\}_{i=1}^{n_{\mathcal{C}}}$, modeled by curves that correspond to their mean line. In this case, the vector μ may contain physical coefficients of the constitutive equations of the steel or the concrete. In view of identifying the displacements in each subdomain, we shall note u_μ^{c} the displacement in the concrete, and u_μ^{s} the displacement in the steel. Both of those fields can be seen as restrictions of u_μ on the corresponding domain. We should distinguish in a similar manner between deformations, stresses or internal variables in both computational domains. On the one hand, the mechanical strains tensor within the concrete is the symmetric gradient of the displacement and is denoted $\varepsilon_\mu^{\mathrm{c}} = \nabla_s u_\mu^{\mathrm{c}} = \frac{1}{2}\left(\nabla u_\mu^{\mathrm{c}} + \left(\nabla u_\mu^{\mathrm{c}}\right)^{\top}\right)$; on the other hand, the strains within the cables (also called uniaxial strains) are defined as $\varepsilon_\mu^{\mathrm{s}} = \partial_s u_\mu^{\mathrm{s}}$, where $\partial_s(.)$ is the derivative along the cable. As for the internal forces, we denote the stress tensor within the concrete σ_μ, the normal forces in the steel N_μ, and the internal variables in the concrete γ_μ^{c} and in the steel γ_μ^{s}. The heterogeneous nature of the material, which is broken down into several zones, explains our decision to label this case as a multi-modeling scenario.

For more detailed behavior modeling, we enable external phenomena to be accounted for in the mechanical behavior equations. Specifically, the mechanical state of concrete evolves as a function of its temperature or degree of hydration. As a result, rheological phenomena are typically thermo-hydro-activated. In other words, we assume that the constitutive equations used depend on auxiliary variables, which we shall refer to as a vector H. The fields enclosed in H include previously computed fields and solutions to PDEs that do not depend on the parameters set in the vector μ. The latter is comprised of fields that may appear and be used in the problem's constitutive or evolution equations. In the application case presented, namely in the case of a thermo-hydro-activated mechanical problem, this vector consists of the pair made of temperature and water content in the concrete. We introduce the quasi-static equilibrium equations for the three-dimensional model, where we omit to specify the initial conditions (ICs) and the BCs for each subdomains:

$$\left\{\begin{array}{rcl} -\nabla \cdot \sigma_\mu^{\mathrm{c}} &=& f_{\mathrm{c}} \quad \text{on } \Omega^{\mathrm{c}}, \\[2ex] \sigma_\mu^{\mathrm{c}} &=& \mathcal{F}_\mu^{\sigma}\left(\varepsilon_\mu^{\mathrm{c}},\ \gamma_\mu^{\mathrm{c}},\ H\right), \\[2ex] \dot{\gamma}_\mu^{\mathrm{c}} &=& \mathcal{F}_\mu^{\gamma^{\mathrm{c}}}\left(\sigma_\mu^{\mathrm{c}}, \gamma_\mu^{\mathrm{c}},\ H\right), \end{array}\right. \quad \text{and} \quad \left\{\begin{array}{rcl} \dfrac{\partial \mathrm{N}_\mu}{\partial s} &=& f_{\mathrm{s}} \quad \text{on } \Omega^{\mathrm{s}}, \\[2ex] \mathrm{N}_\mu &=& \mathcal{F}_\mu^{\mathrm{N}}\left(\partial_s u_\mu^{\mathrm{s}},\ \gamma_\mu^{\mathrm{s}},\ H\right), \\[2ex] \dot{\gamma}_\mu^{\mathrm{s}} &=& \mathcal{F}_\mu^{\gamma^{\mathrm{s}}}\left(\mathrm{N}_\mu, \gamma_\mu^{\mathrm{s}},\ H\right), \end{array}\right.$$

where $\mathcal{F}_\mu^{\sigma}$ (resp. $\mathcal{F}_\mu^{\mathrm{N}}$) stands for the constitutive equation for the three-dimensional (resp. one-dimensional) problem, while the nonlinear operator $\mathcal{F}_\mu^{\gamma^{\mathrm{c}}}$ (resp. $\mathcal{F}_\mu^{\gamma^{\mathrm{s}}}$) denotes an equation of evolution of internal variables within the concrete (resp. the steel). Further details on the models used in practice and the dependencies in auxiliary variables are given in Chapter 3.

We seek to possess a generic formulation for the multi-modeling problem, enabling us to align it with a problem whose formulation closely mirrors the one previously introduced, specifically in the context of single-modeling. Attaining common notations subsequently facilitates the availability of shared operators, applicable in a general manner to both introduced problems. To that end, we introduce new notations for the fields defined on the whole domain, namely for the displacements, strains, generalized forces (stresses or normal efforts), internal variables and the loadings. All the details are provided in Table 2.1.

Notation on Ω	**Notation on Ω^{s}**	**Notation on Ω^{c}**	**Definition**
$\mathfrak{S}_\mu$	N_μ	σ_μ	Generalized force
u_μ	u_μ^{s}	u_μ^{c}	Displacement
ε_μ	$\varepsilon_\mu^{\mathrm{s}} = \partial_s u_\mu^{\mathrm{s}}$	$\varepsilon_\mu^{\mathrm{c}}$	Strain
γ_μ	γ_μ^{s}	γ_μ^{c}	Internal variables
f	f_{c}	f_{s}	External loading

Table 2.1: Notations of the fields defined on the whole computational domain Ω, whose definition depends on the subdomains (Ω^{c} or Ω^{s}).

These notations enable us to recast the problem in a compact form, which helps to manage the multi-modeling framework (3d-1d) using three operators, $\mathcal{G}_\mu\left(.\right)$ for the equilibrium equation, $\mathcal{F}_\mu^{\mathfrak{S}}\left(.\right)$ for the constitutive equation and $\mathcal{F}_\mu^{\gamma}\left(.\right)$ for the evolution equation for internal variables:

$$\left\{\begin{array}{rcl} \mathcal{G}_\mu\left(\mathfrak{S}_\mu\right) & = & f, \\ \mathfrak{S}_\mu & = & \mathcal{F}_\mu^{\mathfrak{S}}\left(\mathfrak{S}_\mu,\ \gamma_\mu,\ H\right), \\ \dot{\gamma}_\mu & = & \mathcal{F}_\mu^{\gamma}\left(\varepsilon_\mu,\ \gamma_\mu,\ H\right), \end{array}\right.$$

where we still omit the ICs and BCs used. In our study, the initial state of the problem is the material at rest, so all physical fields are assumed to be zero initially. Since a one-step time integrator is used (cf. Equation (1.3)), it implies that the knowledge of the mechanical state is derived from the mechanical state previously computed and the knowledge of the field H at the current time. In our context of multi-modeling problems, BCs reflect engineering assumptions designed to provide a realistic simulation. The latter comprise both non-homogeneous Neumann conditions (defined on $\Gamma_{\mathrm{n}}^{\mathrm{c}}$ for the concrete) and homogeneous Dirichlet conditions for suitable linear combinations of the state variables. As in the preceding case, we assume that the displacement field belongs to the kernel of this form (c linear form in Equation (2.9)). In the general framework of the unidimensional problem, Neumann BCs on a given cable $\mathcal{C}_i$ are expressed as application of nodal forces $F_{i,j}$ applied on a set of discrete points $\{x_j^{\mathcal{C}_i}\}_{j=1}^{n_{\mathcal{C}_i}^{1\mathrm{d}}}$. This translates into a jump $[\![.]\!]$ in the normal efforts at every point $x_j^{\mathcal{C}_i}$. In the end, the multi-modeling problem can be written as:

$$\left\{\begin{array}{rcll} \mathcal{G}_\mu\left(\mathfrak{S}_\mu^{(k)}\right) & = & f^{(k)} & \text{on} \quad \Omega, \\ \mathfrak{S}_\mu^{(k)} & = & \mathcal{F}_\mu^{(k)}\left(u_\mu^{(k)},\ u_\mu^{(k-1)}, \mathfrak{S}_\mu^{(k-1)},\ H^{(k)}\right) & \text{on} \quad \Omega, \end{array}\right. \tag{2.8}$$

with BCs expressed as follows:

$$\begin{cases} \text{Dirichlet BCs :} \quad c(u_\mu^{(k)}) = 0 \text{ on } \Omega, \\ \\ \text{Neumann BCs :} \quad \begin{cases} (\sigma_\mu)^{(k)} \cdot n &= f_s^{(k)} & \text{on} & \Gamma_{\mathrm{n}}^{\mathrm{c}}, \\ [\![\mathrm{N}_\mu^{(k)}]\!](x_j^{\mathcal{C}_i}) &= F_{i,j}^{(k)} \;\; \forall j \in \{1, ..., n_{\mathcal{C}_i}^{1\mathrm{d}}\} & \text{for} & \mathcal{C}_i, \; \forall i \in \{1, ..., n_{\mathcal{C}}\}, \end{cases} \end{cases} \tag{2.9}$$

Eventually, the multi-modeling problem written in compact form in the Equation (2.8) to which the BCs are applied lead to the following variational problem $\forall k \in \{1, ..., K\}$:

$$\text{Find } u_\mu^{(k)} \in \mathcal{X}_{\mathrm{bc}} \text{ s.t.} \begin{cases} \mathcal{R}_\mu \left(u_\mu^{(k)}, \; u_\mu^{(k-1)}, \; \mathfrak{S}_\mu^{(k)} \right) = 0, & \forall v \in \mathcal{X}_{\mathrm{bc}}, \\ \mathfrak{S}_\mu^{(k)} = \mathcal{F}_\mu^{(k)} \left(u_\mu^{(k)}, \; u_\mu^{(k-1)}, \mathfrak{S}_\mu^{(k-1)}, \; H^{(k)} \right) \;\; \text{on} \;\; \Omega, \end{cases} \tag{2.10}$$

where $\mathcal{X}_{\mathrm{bc}}$ is defined in the same manner as in the single-modeling case. Indeed, the provided formulation of the boundary conditions is consistent across all given formulations. We denote:

$$\mathcal{R}_\mu \left(u_\mu^{(k)}, \; u_\mu^{(k-1)}, \; \mathfrak{S}_\mu^{(k)} \right) = \mathcal{R}_\mu^{\mathfrak{S}} \left(\mathcal{F}_\mu^{(k)} \left(u_\mu^{(k)}, \; u_\mu^{(k-1)}, \mathfrak{S}_\mu^{(k-1)}, \; H^{(k)} \right), \; v \right), \text{ and } \mathcal{R}_\mu^{\mathfrak{S}} \left(\mathfrak{S}, \; v \right) = \begin{bmatrix} \mathcal{R}_\mu^{\sigma} \left(\sigma_\mu^{(k)}, \; v \right) \\ \mathcal{R}_\mu^{\mathrm{N}} \left(\mathrm{N}_\mu^{(k)}, \; v \right) \end{bmatrix},$$

where we introduce the notations $\forall v \in [v^{\mathrm{c}}, \; v^{\mathrm{s}}]^\top$:

$$\begin{cases} \mathcal{R}_\mu^{\sigma} \left(\sigma_\mu^{(k)}, \; v \right) &= \displaystyle\int_\Omega \sigma_\mu^{(k)} : \varepsilon \left(v^{\mathrm{c}} \right) \, d\Omega - \int_\Omega f_v \cdot v^{\mathrm{c}} \, d\Omega - \int_\Gamma f_s \cdot v^{\mathrm{c}} \, d\Gamma, \\ \\ \mathcal{R}_\mu^{\mathrm{N}} \left(\mathrm{N}_\mu^{(k)}, \; v \right) &= \displaystyle\int_{\mathcal{C}} \mathrm{N}_\mu^{(k)} : \partial_s v^{\mathrm{s}} \, ds - \int_{\mathcal{C}} f_v \cdot v^{\mathrm{s}} \, ds - \sum_{i=1}^{n_{\mathcal{C}}} \sum_{j=1}^{n_{\mathcal{C}_i}^{1\mathrm{d}}} F_{i,j}^{(k)} v^{\mathrm{s}}(x_j^{\mathcal{C}_i}). \end{cases}$$

In the residuals presented here, there are two residuals, one associated with the three-dimensional model, similar to the one-dimensional case, and one associated with the one-dimensional model. We can observe that both formulations, i.e. single-modeling (cf. Equation (2.6)) and multi-modeling (cf. Equation (2.10)), are consistent. The variational formulation can be articulated in a comparable fashion for both, with similar treatment of boundary conditions. Moreover, the multi-modeling scenario can be regarded as an extension of the single-modeling case, where the vector $\boldsymbol{\mathfrak{S}}$ is confined to stresses $\boldsymbol{\sigma}$.

2.2.2 High-fidelity discretization

Given the domain $\Omega \subset \mathbb{R}^d$, we consider a HF mesh $\mathcal{T}^{\mathrm{hf}} = \{\mathtt{D}_i\}_{i=1}^{N_{\mathrm{e}}}$ where $\mathtt{D}_1, \ldots, \mathtt{D}_{N_{\mathrm{e}}}$ are the elements of the mesh, and N_{e} denotes the number of elements in the mesh. In the introduced notations, we allow for meshes with multiple types of elements within the same mesh. This feature is closely tied to the architecture of the FE code employed. In this code, specifically for a three-dimensional mesh, a mesh is structured with multiple levels, incorporating surface elements to handle applied surface forces effectively. Moreover, for the targeted industrial application, a coupling between three-dimensional elements and one-dimensional elements requires the presence of different types of elements (in terms of dimension and possibly order) to ensure an appropriate formulation. The integer N_{e} denotes the total number of three-dimensional, two-dimensional or one-dimensional elements. Besides, we introduce the continuous Lagrangian finite element (FE) basis $\{\varphi_i\}_{i=1}^{\mathcal{N}_{\text{no}}}$ associated with the mesh $\mathcal{T}^{\text{hf}}$, whose total number of nodes is $\mathcal{N}_{\text{no}}$. The FE space for the primal unknown is thus defined as follows:

$$\mathcal{X}^{\text{hf}} := \operatorname{span}\{\varphi_i e_j,\ i \in \{1, ..., \mathcal{N}_{\text{no}}\}, j \in \{1, ..., d\}\} \tag{2.11}$$

where $e_1, ..., e_d$ are the vectors of the canonical basis. We further define the nodes $\{x_i^{\text{hf,no}}\}_{i=1}^{\mathcal{N}_{\text{no}}}$, the quadrature points $\{x_i^{\text{hf,qd}}\}_{i=1}^{\mathcal{N}_{\text{qd}}}$ associated to the HF-mesh and to the FE discretization and the application $T^{\text{hf,no}}$ (resp. $T^{\text{hf,qd}}$ for the quadrature points) which links the global indexing of the DOFs (resp. unknowns at quadrature points) of the HF-mesh to the local indexing of a specified element. The i_{loc}-th DOF in the q-th element local indexing is associated to the i_{glob} DOF in the global indexing:

$$T^{\text{hf,no}}\left(i_{\text{loc}}, q\right) = i_{\text{glob}}, \quad i_{\text{loc}} = 1, \ldots, n_{\text{lp}}^{q} \text{ and } q = 1, \ldots, N_{\text{e}}$$

where n_{lp}^{q} is the number of DOFs in the q-th element of the mesh. To further clarify the notation, we denote by $\mathbf{u} \in \mathbb{R}^{\mathcal{N}}$ the FE discrete vector of displacements, where $\mathcal{N} = d\mathcal{N}_{\text{no}}$ is the dimension of the space $\mathcal{X}^{\text{hf}}$ in the single-modeling case, and $\mathcal{N} = 3\left(\mathcal{N}_{\text{no}}^{\text{3d}} + \mathcal{N}_{\text{no}}^{\text{1d}}\right)$ in the multi-modeling case, where $\mathcal{N}_{\text{no}}^{\text{3d}}$ (resp. $\mathcal{N}_{\text{no}}^{\text{1d}}$) is the number of nodes for the concrete (resp. steel). Indeed, as a reminder, in the industrial case of prestressed concrete, we have a mesh for the concrete and another for the steel bars, with each node in both meshes having three degrees of freedom ($\Omega \subset \mathbb{R}^3$). As far as quantities at quadrature points are involved, we shall distinguish between two cases, depending on the two aforementioned scenarios. In the case of single-modeling (the purely three-dimensional case, for example), we have the stress vector $\boldsymbol{\sigma} \in \mathbb{R}^{\mathcal{N}_g}$, with $\mathcal{N}_g = \frac{d(d+1)}{2}\mathcal{N}_{\text{qd}}$, while in the multi-modeling case, the generalized forces within the material are denoted by $\boldsymbol{\mathfrak{S}}_\mu = [\boldsymbol{\sigma}_\mu,\ \mathbf{N}_\mu]^\top \in \mathbb{R}^{\mathcal{N}_g}$, since they are unknowns at quadrature points. For the record, the size of these vectors is $\mathcal{N}_g = \mathcal{N}_g^{\text{3d}} + \mathcal{N}_g^{1d} = 6\mathcal{N}_{\text{qd}}^{\text{3d}} + \mathcal{N}_{\text{qd}}^{1d}$, where $\mathcal{N}_{\text{qd}}^{\text{3d}}$ stands for the number of quadrature weights used for the three-dimensional mesh and $\mathcal{N}_{\text{qd}}^{1d}$ for the one-dimensional mesh.

Referring back to the remark in the preceding section, we specifically address the discretization in the multi-modeling case, since the variational form for the single-modeling case is similar. We denote by $T^{\text{hf,no}}$ the connectivity matrix. In the perspective of a hyper-reduced formulation introduced in Section 2.4.2, we need to introduce two elemental restriction operators: the nodal elemental restriction operators $\{\mathbf{E}_q^{\text{no}}\}_{q=1}^{N_{\text{e}}}$ which restrict fields defined at nodes to the q-th element of the mesh (e.g. for displacements) and the quadrature restriction operators $\{\mathbf{E}_q^{\text{qd}}\}_{q=1}^{N_{\text{e}}}$, which handle fields defined at quadrature nodes (e.g., stresses/generalized forces and internal variables):

$$\left(\mathbf{E}_q^{\text{no}}\mathbf{u}\right)_{i_{\text{loc}}} = (\mathbf{u})_{T^{\text{hf,no}}(i_{\text{loc}},q)} \qquad \text{and} \qquad \left(\mathbf{E}_{q'}^{\text{qd}}\boldsymbol{\mathfrak{S}}\right)_{j_{\text{loc}}} = (\boldsymbol{\mathfrak{S}})_{T^{\text{hf,qd}}(j_{\text{loc}},q')}$$

We denote by $\{\mathbf{u}_\mu^{\text{hf},(k)}\}_{k=1}^{K}$ the FE approximation of the displacement (primal variable) given by the HF-model at all times, whereas $\{\boldsymbol{\mathfrak{S}}_\mu^{\text{hf},(k)}\}_{k=1}^{K}$ stand for the generalized force fields (stress or normal efforts). We state the FE discretization of the variational form defined by Equation (2.10), $\forall k \in \{1, ..., K\}$, find $u_\mu^{\text{hf},(k)} \in \mathcal{X}_{\text{bc}}^{\text{hf}}$ such that:

$$\left\{\begin{aligned} \mathcal{R}_\mu^{\text{hf}}\left(\mathbf{u}_\mu^{\text{hf},(k)},\ \mathbf{u}_\mu^{\text{hf},(k-1)},\ \boldsymbol{\mathfrak{S}}_\mu^{\text{hf},(k-1)}, \mathbf{v}\right) &= 0, & \forall v \in \mathcal{X}_{\text{bc}}^{\text{hf}}, \\ \boldsymbol{\mathfrak{S}}_\mu^{\text{hf},(k)} &= \mathcal{F}_\mu^{\text{hf}}\left(\mathbf{u}_\mu^{\text{hf},(k)}, \mathbf{u}_\mu^{\text{hf},(k-1)},\ \boldsymbol{\mathfrak{S}}_\mu^{\text{hf},(k-1)},\ \mathbf{H}^{\text{hf},(k)}\right), & \end{aligned}\right. \tag{2.12}$$

where $\mathcal{X}_{\text{bc}}^{\text{hf}} := \left\{\mathbf{v} \in \mathcal{X}^{\text{hf}} : \quad \mathbf{B}\mathbf{v} = 0\right\}$ depicts the test space for displacements, and $\mathbf{B} \in \mathbb{R}^{\mathcal{N}_d \times \mathcal{N}}$ is the kinematic relationship matrix. $\mathcal{N}_d$ stands for the number of linear relations between degrees of freedom that we intend to enforce. This matrix reflects the discretization of the linear form formulated by Equation 2.4. Such a formulation on the BCs implies that the kinematic linear application depends neither on time nor on the parameter. Each line reflects a kinematic relationship between nodes of the overall mesh. Therefore, in the multi-modeling case, the said matrix includes not only the Dirichlet BCs applied to each physical domain, but also the kinematic relationships between the nodes of two distinct models (kinematic coupling). The independence of the matrix vector in the parameter (and the displacement field) implies in particular that we do not deal here with instances of sliding between nodes. In the multi-modeling case we shall discuss, this means that there is no tendon sliding in the concrete. The operators $\mathcal{R}_\mu^{\mathrm{hf}}$ and $\mathcal{F}_\mu^{\mathrm{hf}}$ stands for the discrete counterparts of the continuous operators $\mathcal{R}_\mu$ and $\mathcal{F}_\mu$ introduced in Equation (2.12). The practical implementation of the FE method relies on iterating through all elements, successively calling local assemblies to aggregate the elemental contributions. We emphasize that the assembly procedure can be split into several terms. For example, in the case of the multi-modeling framework, namely for prestressed concrete, the assembly will be split into loop for concrete elements (three-dimensional for constitutive equations and volumic loadings, and two-dimensional for surfacic loadings), and a loop for the steel-elements (one-dimensional for the constitutive equation and loadings). Thus, in practice, the FE code compute the HF-residuals as sums of elementary contributions, which can be formulated as follows, $\forall v \in \mathcal{X}^{\mathrm{hf}}$:

$$
\begin{aligned}
&\mathcal{R}_\mu^{\mathrm{hf}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\mathfrak{S}}_\mu^{(k-1)},\ \mathbf{v}\right) = \sum_{q=1}^{N_e} \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\mathfrak{S}}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right) \\
&= \underbrace{\sum_{q=1}^{N_e^{\mathrm{3d}}} \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd,3d}}\boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right)}_{:=\mathcal{R}_\mu^{\mathrm{hf,3d}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{v}\right)} + \underbrace{\sum_{q=1}^{N_e^{\mathrm{2d}}} \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd,2d}}\boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right)}_{:=\mathcal{R}_\mu^{\mathrm{hf,2d}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{v}\right)} \\
&+ \underbrace{\sum_{q=1}^{N_e^{\mathrm{1d}}} \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd,1d}}\mathbf{N}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right)}_{:=\mathcal{R}_\mu^{\mathrm{hf,1d}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \mathbf{N}_\mu^{(k-1)},\ \mathbf{v}\right)},
\end{aligned}
$$

where $\mathbf{E}_q^{\mathrm{no}}$ (resp. $\mathbf{E}_q^{\mathrm{qd}}$) is an elementary restriction operator on vectors at nodes (resp. quadrature points). For operators on vectors at quadrature points, we adopt the specific notation $\mathbf{E}_q^{\mathrm{qd},\bullet}$ for $\bullet \in \{\mathrm{1d}, \mathrm{2d}, \mathrm{3d}\}$.

To adhere to the theoretical framework required by the formulations in our HF setting, we handle Dirichlet BCs by dualizing them through the introduction of Lagrange multipliers. Within this context, the vector solution of the problem at the k-th timestep comprises the displacements and the associated Lagrange multipliers $(\mathbf{u}_\mu^{(k)}, \boldsymbol{\lambda}_\mu^{(k)}) \in \mathbb{R}^{\mathcal{N}} \times \mathbb{R}^{\mathcal{N}_d}$. The assembly of the FE problem results in the discrete nonlinear system:

$$
\left\{
\begin{array}{rcl}
\mathbf{R}_\mu^{\mathrm{hf}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\mathfrak{S}}_\mu^{(k-1)}\right) + \mathbf{B}^\top\boldsymbol{\lambda}_\mu^{(k)} & = & \mathbf{0}, \\
\mathbf{B}\mathbf{u}_\mu^{(k)} & = & \mathbf{0}.
\end{array}
\right.
\tag{2.13}
$$

We employ the Newton-Raphson algorithm to solve (2.13) [EDF24]. It is worth noting that the Jacobian of the FE discrete residual (2.13) takes the form of a saddle-point system. Dualizing Dirichlet BCs provides a natural framework for enforcing Dirichlet-type conditions in the interior of the domain and/or at points that do not coincide with the nodes of the mesh. Readers interested in the process of solving the discretized system in the case of dualization can refer to Appendix A.

2.3 Reduced space construction

In this section, our discussion covers the algorithms needed for constructing a reduced basis for the problems under investigation. As previously stated, the construction of a reduced basis starts with a set of precomputed HF solutions, called snapshots, from which we shall construct a low-dimensional space approximating the solutions sufficiently accurately. We leave the notion of the approximation quality achieved for a given reduced space for later. In detailing the principle of the reduced basis method, two challenges naturally arise. First, we must know how to choose the initial HF solutions, and therefore how to determine the parameters we need to use for training in $\mathcal{P}$. Therefore, the first question is then to devise a sampling strategy in the parameter space. The second issue concerns the generation of a reduced basis given a set of available snapshots.

2.3.1 Reduced basis approach

Employing numerical methods requires dealing with discrete sets, entailing an initial discretization Θ_{train} of the parameter space $\mathcal{P}$. This discrete set is often referred to as a training set. Typically, a Cartesian grid discretization is employed in standard procedures. However, sampling the space in this manner and generating snapshots for all parameters can lead to exceptionally high computational costs, particularly when dealing with high-dimensional parametric spaces. Alternative methodologies, such as space-filling designs for uncertainty quantification like Latin Hypercube Sampling [MBC00], can be employed. The initial sampling of the parameter space constitutes a research topic on its own, offering a multitude of possibilities that heavily depend on the user's final intention or the nature of the physical problem. There is no inherent evidence suggesting that the physical problem exhibits regularity in the parameter space, justifying Cartesian grid discretization. To avoid a priori reliance on Cartesian grid discretization, efforts have been directed toward the philosophy of mesh refinement. These methods involve intricate mathematical tools, including a metric on approximation spaces, such as the Grassmann distance [BI19]. In the context of the presented work, however, we opted for an initial discretization on a Cartesian grid for simplicity's sake, focusing on other pertinent issues. The studied scenarios are carried out on low-dimensional parametric spaces, which justifies this choice. In such scenarios, greedy approaches are often employed to iteratively generate a set of snapshots. This entails having two sets: one comprises the previously discrete set of candidates, that is the training set Θ_{train}, while the other is constructed iteratively by identifying the appropriate parameters to explore from the previous set. Although initially designed for static cases, this approach can be adapted to time-dependent problems. The details of the greedy procedure will be expounded upon in the subsequent sections. Once all parameters are acquired, we effectively obtain the associated HF simulations and consequently the solutions to these problems.

As mentioned earlier, we operate within the framework of linear approximations. We seek the reduced-order solution as a linear combination of modes:

$$\widehat{\mathbf{u}}_{\mu}^{(k)} = \sum_{n=1}^{N_u} \left(\widehat{\alpha}_{u,\mu}^{(k)}\right)_n \boldsymbol{\zeta}_{u,n} = \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{u,\mu}^{(k)} \tag{2.14}$$

where $\widehat{\boldsymbol{\alpha}}_{u,\mu}^{(k)} \in \mathbb{R}^{N_u}$ are referred to as generalized coordinates (or reduced coordinates) and $\mathcal{Z}_{N_u} = \operatorname{span}\{\zeta_{u,n}\}$ is the primal reduced space. The reduced space of dimension N_u arising from the base vectors constitutes the new approximation space in which solutions are sought (cf. Figure 2.1). To ensure that the method delivers computational benefits, this space must be designed to be of lower dimension than the HF space, i.e. $N_u \ll \mathcal{N}$.

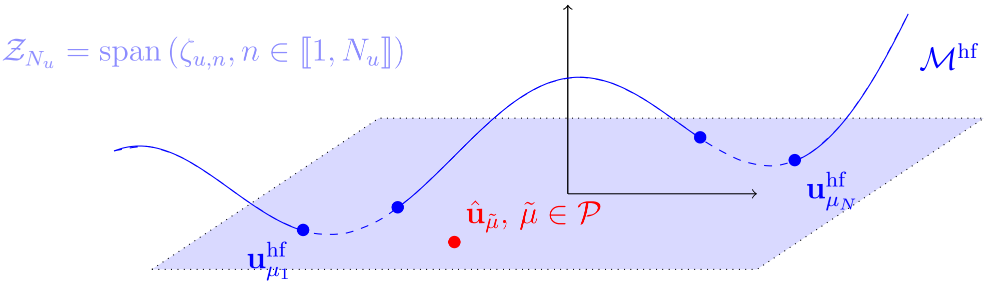


Figure 2.1: A visualization example of a linear reduced space acting as an approximation space to explore the parametric manifold $\mathcal{M}^{\text{hf}} = \{u_\mu,\ \mu \in \mathcal{P}\}$. Adaptation of Figure from [Qua17].

One popular method that often yields good results is Proper Orthogonal Decomposition (POD [BHL93][BBI09][Vol11]): it involves choosing a subspace that optimally describes a given set of data. This method goes by different names in different communities (e.g., principal component analysis, or Karhunen-Loève expansion). Other approaches exist for specific problems, especially in the realm of linear control problems, where balanced truncation [Moo81] approaches or dynamic mode decomposition [Sch10] are popular model reduction procedures. However, in our case, we choose to rely on the POD method to construct our low-dimensional approximation space. The subsequent paragraphs (see Section 2.3.3) cover the principles and computational details of the method.

The approach hinges upon the construction of a linear space tailored for approximating the parametric manifold of solutions, denoted as $\mathcal{M}^{\text{hf}} = \{u_\mu,\ \mu \in \mathcal{P}\}$. From a theoretical standpoint, assessing the potential quality of approximation necessitates the application of tools derived from the approximation theory. In order to estimate the accuracy of approximating the manifold $\mathcal{M}^{\text{hf}}$ using a low-dimensional linear space $\mathcal{V}_N$, we introduce the Kolmogorov N-width $d_N\left(\mathcal{M}^{\text{hf}}, \mathcal{X}\right)$. For a given finite dimensional space $\mathcal{V}_N$, we measure the deviation of $\mathcal{M}^{\text{hf}}$ from $\mathcal{V}_N$ as follows:

$$\operatorname{dist}_{\mathcal{X}}\left(\mathcal{M}^{\text{hf}}, \mathcal{V}_N\right) = \sup_{u_\mu \in \mathcal{M}^{\text{hf}}} \inf_{v_N \in \mathcal{V}_N} \|u_\mu - v_N\|_{\mathcal{X}}$$

Consequently, the Kolmogorov N-width gauges the manifold's amenability to approximation by an optimally chosen vectorial space of dimension N:

$$d_N\left(\mathcal{M}^{\text{hf}}, \mathcal{X}\right) = \inf_{\mathcal{V}_N \subset \mathcal{X}^{\text{hf}},\ \dim(\mathcal{V}_N) = N} \operatorname{dist}_{\mathcal{X}}\left(\mathcal{M}^{\text{hf}}, \mathcal{V}_N\right)$$

We assume that this quantity decays fast with N in the cases considered in our work. Introduced by Kolmogorov [Kol], this metric delineates the best achievable accuracy when approximating all conceivable elements of $\mathcal{M}^{\text{hf}}$ from an N-dimensional linear space $\mathcal{V}_N$, according to the $\mathcal{X}$ norm. A swift decay indicates that, for a given level of accuracy, an approximate space of modest size can be constructed. The classical problem of approximating any element u_μ belonging to $\mathcal{M}^{\text{hf}}$ through finite expansions has been extensively addressed, with reduced basis methods among the prominent techniques [MPT02][CD15][CD16]. The regularity of u_μ in μ is identified as a potential factor contributing to a small N-width, as indicated by prior research findings. Specifically, RB methods, when equipped with well-selected bases, prove adept at capturing additional information pertinent to the underlying problem. The iterative construction of a linear approximation space involves the careful selection of particular snapshots. Despite the non-optimality of these generated spaces, prior investigations suggest that a specific greedy recursive parameter selection within the RB method yields convergence rates comparable to those achieved by optimal algorithms for the approximation space. Furthermore, model reduction techniques such as the generalized empirical interpolation method (GEIM [MM13]) or the generalized reduced basis method [LMQR13] demonstrate a decay in the Kolmogorov width of the solution manifold, closely aligning with d_N. Some classes of physical problems exhibit a gradual decrease in the Kolmogorov N-width, with convection-dominated issues being a common example in this category. Though methods exist to address this issue, discussions on these solutions fall outside the scope of this contribution, since the focus of this work does not encompass the investigation of slow decays.

Within our work, the justification for a linear approximation lies in the inherent regularity of the target problem (creep behavior of prestressed concrete while ageing), which renders a linear approach both viable and efficient. It is essential to note, however, that this choice is situated within a landscape abundant with an alternative category of methods known as nonlinear approximation methods, which do not rely on approximations from linear spaces but rather from nonlinear manifolds (the application of these methods exceeds the scope of this thesis). Nonlinear approaches founded on piecewise affine spaces have been first explored [AZF12][WAZF12][AZW15][GFTBM21]. Thus, other methodologies often leveraging machine learning or deep-learning tools to craft nonlinear approximation spaces have been introduced. They can be cluster-based in order to associate a proper local basis (ROM-net [DCAR20][PÁKRR21]), or adopt totally non-intrusive approaches employing neural networks [FDM21][FM22]. Another approach involves the complementary use of nonlinear mappings to enhance a linear approximation, for example by relying on neural-networks [BFM23]. It is important to note that our presentation of those methodologies is not exhaustive.

2.3.2 POD-Greedy approach as an iterative algorithm for sampling the parameter space

Greedy algorithms have emerged as efficient tools in the field of model reduction, offering a systematic approach to build ROMs by iteratively selecting the most influential parameters. The key to this approach is to determine such parameters out of a given set of parameters. To elucidate the greedy process, we first focus on the case of a static problem. Within the realm of greedy algorithms, several categories of methodologies exist, namely strong greedy and weak-greedy approaches. Originally, strong greedy methods were developed for static problems. These approaches hinge on the identification of the best fit error ($E^{\mathrm{bf}}_{\mu,N}$), corresponding to the projection error that is the norm of the component outside the approximation space (reduced space):

$$E^{\mathrm{bf}}_{\mu,N} = \left\| u^{\mathrm{hf}}_\mu - \Pi_{\mathcal{Z}_N} u^{\mathrm{hf}}_\mu \right\|_{\mathcal{X}}, \quad \text{with} \quad \Pi_{\mathcal{Z}_N} u^{\mathrm{hf}}_\mu = \arg\min_{v \in \mathcal{Z}_N} \left\| u^{\mathrm{hf}}_\mu - v \right\|_{\mathcal{X}}$$

where $\Pi_{\mathcal{Z}_N} : \mathcal{X} \to \mathcal{X}$ stands for the orthogonal projection onto the reduced space $\mathcal{Z}_N$. The objective is to pinpoint the parameter least accurately approximated by the generated space, i.e., the one with the largest projection error:

$$\mu^* = \arg\min_{\mu \in \Theta_{\mathrm{train}}} E^{\mathrm{bf}}_{\mu,N}$$

At each greedy iteration, this approach relies on computing the best-fit errors on a discrete parameter set in order to identify a parameter μ^* for which the error is large. It resorts to a HF solver to estimate the solution for μ^*, and then it updates the reduced-order basis by using the new snapshot and modifies accordingly the ROM structures. Thus, greedy procedures are inherently sequential.

These approaches were subsequently extended to unsteady problems. In this scenario, the solutions define a trajectory, characterized as the set of snapshots at each time step $\mathbb{U}_\mu = \{u_\mu^{(k)}\}$. The crucial aspect involves devising a novel method for gauging the discrepancy between two solutions and their temporal evolution. Presented in Equations (2.15) (2.16) (2.17) are various illustrative examples applicable contingent upon the specific case. It is imperative to highlight that the judicious selection of a method hinges on the nature of the problem under investigation and the precise quantities we wish to approximate, whether it be the final state or the entire trajectory, for instance. For instance, one can consider the projection error in the final state (cf. Equation (2.15)), the maximum projection error over all time steps (cf. Equation (2.16)), or an error on the trajectories defined with a specific metric (cf. Equation (2.17)). We do not delve into the detailed definition of this metric, which can be chosen to consider a time-averaged norm, for example, using an $L^2(0, t_\mathrm{f})$ norm.

$$\mu^* = \arg \min_{\mu \in \Theta_\mathrm{train}} \left\| u_\mu^{(K)} - \Pi_{\mathcal{Z}_N} u_\mu^{(K)} \right\|_{\mathcal{X}} \tag{2.15}$$

$$\mu^* = \arg \min_{\mu \in \Theta_\mathrm{train}} \left(\max_{k \in \{0,\dots,K\}} \left\| u_\mu^{(k)} - \Pi_{\mathcal{Z}_N} u_\mu^{(k)} \right\|_{\mathcal{X}} \right) \tag{2.16}$$

$$\mu^* = \arg \min_{\mu \in \Theta_\mathrm{train}} |||\mathbb{U}_\mu - \Pi_{\mathcal{Z}_N} \mathbb{U}_\mu|||_{\mathcal{X}} \tag{2.17}$$

$$\tag{2.18}$$

Research on the theoretical convergence properties of these strong-greedy algorithms has been conducted for steady-state problems [BCD$^+$11][BMP$^+$12]. Specifically, it has been demonstrated that for problems characterized by an exponentially small Kolmogorov N-width, one can achieve convergence results between the obtained reduced solution and the HF solution. This implies an exponential convergence of the approximation error. The fundamental concept underpinning these algorithms is to establish the theoretical appropriateness of relying on greedy algorithms for constructing reduced models. These findings have also been extended to time-dependent problems [Haa13] in subsequent studies, showcasing the validity of applying these greedy approaches to the problems addressed in this thesis.

However, strong-greedy algorithm is not an efficient way of building iteratively a ROM since we need to compute HF snapshots to estimate approximation errors. To circumvent this issue, the weak-greedy algorithm, first proposed in [VPRP03] and then analyzed in a series of papers [CD15], relies on *a posteriori* error indicator instead of the approximation error. Thus, since evaluating such error indicator can be cost-efficient, we can overcome the bottleneck of estimating the poorly-approximated solution on a given discrete parameter set. Initially introduced for steady problems, it has been extended to unsteady PDEs [HO08]. Weak-greedy algorithms rely on the definition of an indicator that increases the approximation error (or is correlated with it) in order to be able to drive the greedy process:

$$|||\mathbb{U}_\mu - \Pi_{\mathcal{Z}_N} \mathbb{U}_\mu|||_{\mathcal{X}} \leq \Delta_\mu, \quad \forall \mu \in \mathcal{P} \tag{2.19}$$

Besides, one prominent use of greedy algorithms is in the context of POD and Galerkin projection [Haa13]. These methods aim to capture the dominant modes of a system and create a reduced subspace that preserves the essential dynamics. Greedy algorithms enhance this process by intelligently selecting parameter samples to construct an accurate and compact reduced model. The flexibility of greedy algorithms allows their adaptation to different types of problems, such as time-dependent or nonlinear systems. Their ability to iteratively refine the ROM by selecting additional parameters, guided by a prescribed error criterion, ensures an optimal trade-off between accuracy and computational cost.

2.3.3 Proper Orthogonal Decomposition

In this section, we describe the Proper Orthogonal Decomposition (POD) method and introduce a method for determining such a basis. We refer to [BHL93], or to more recent works [BBI09][Vol11] for more details. As mentioned earlier, the objective of POD is to reduce input data by retaining only essential information using a small number of vectors. In this section, we describe a peculiar approach to constructing a POD basis, known as the method of snapshots [Sir87]. In scientific computing, POD has been extensively utilized for reducing the dimensionality of large datasets arising from simulations or experimental measurements. This reduction in dimensionality not only facilitates a more efficient storage and computation but also aids in extracting meaningful patterns and trends from the data. In fluid dynamics, for instance, POD has been employed to analyze and model turbulent flows, enabling the identification of coherent structures and essential flow features [BHL93]. This approach aims to build an approximation basis from a set of solutions previously obtained and generated from a training set Θ_{train}. This training set is often regarded as a discretization of the parameter space $\mathcal{P}$. The advantage of this approach lies in generating a set of uncorrelated vectors from the obtained HF snapshots. The modes are generated and stored in order of importance, such that the first modes contain the most information regarding the generated snapshots.

Minimization problem for the Proper Orthogonal Decomposition

More specifically, we assume that we have a set of solution vectors stored in a matrix called the snapshot matrix, defined as follows $\mathbf{S} = [\mathbf{u}_1, ..., \mathbf{u}_{n_{\text{train}}}] \in \mathbb{R}^{\mathcal{N} \times n_{\text{train}}}$. We assume that the size of the discrete vectors, i.e., the number of degrees of freedom $\mathcal{N}$ in the problem, is much larger than the size n_{train} of the training set, that is $\mathcal{N} \gg n_{\text{train}}$. This constitutes the foundational assumption for the computational efficiency of the method of snapshots. We assume that we choose a fixed number ℓ of basis vectors. Subsequently, we shall display a metric to optimally determine this rank. One may choose to produce a orthonormal basis $\{\psi_i\}_{i=1}^{\ell}$ that minimizes the distance between the snapshots and the orthogonal projection onto the reduced subspace generated by this orthonormal basis $\tilde{\mathcal{Z}} = \text{span}(\{\psi_i\}_{i=1}^{\ell})$:

$$\left\| u_j - \sum_{i=1}^{\ell} (u_j,\ \psi_i)_{\mathcal{X}} \psi_i \right\|_{\mathcal{X}}^2 = \| u_j - \Pi_{\tilde{\mathcal{Z}}} u_i \|_{\mathcal{X}}^2 = \| \Pi_{\tilde{\mathcal{Z}}^{\perp}} u_i \|_{\mathcal{X}}^2$$

where $\Pi_{\tilde{\mathcal{Z}}} \ : \ \mathcal{X} \to \mathcal{X}$ stands for the orthogonal projection onto $\tilde{Z}$. Therefore, the derivation of the reduced basis is equivalent to solving the following constrained optimization problem:

$$
\begin{aligned}
&\min_{\psi_1,\ldots,\psi_\ell \in \mathcal{X}^{\mathrm{hf}}} \quad \sum_{j=1}^{n_{\mathrm{train}}} \left\| u_j - \sum_{i=1}^{\ell} (u_j,\ \psi_i)_{\mathcal{X}} \psi_i \right\|_{\mathcal{X}}^2 \\
&\text{subject to} \quad (\psi_j,\ \psi_j)_{\mathcal{X}} = \delta_{i,j},\ 1 \leq i, j \leq \ell
\end{aligned}
\tag{2.20}
$$

where the constraint certifies that the resulting basis is orthonormal in the sense of the desired scalar product $(\cdot,\cdot)_{\mathcal{X}}$. From a discrete viewpoint, the resulting basis is optimal in the sense of the Frobenius norm F_{X} expressed thanks to the induced norm matrix $\mathbf{X}$, such that, $\forall u, v \in \mathcal{X}^{\mathrm{hf}}$, $(u,\ v)_{\mathcal{X}} = \mathbf{u}^\top \mathbf{X} \mathbf{v}$, and defined such that forall $\mathbf{A}$: $\|\mathbf{A}\|_{F_{\mathrm{X}}} = \operatorname{trace}\left(\mathbf{A}^\top \mathbf{X} \mathbf{A}\right)$. By introducing the projection matrix $(\mathbf{P})_{i,j} = (u_j,\ \psi_i)_{\mathcal{X}}$, which gives in a more compact form $\mathbf{P} = \widetilde{\mathbf{Z}}_u^\top \mathbf{X} \mathbf{S}$, the previous optimization problem can be restated in a discrete way as follows:

$$
\begin{aligned}
&\mathbf{Z}_u = \arg \min_{\widetilde{\mathbf{Z}}_u \in \mathbb{R}^{\mathcal{N} \times \ell}} \quad \left\| \mathbf{S} - \widetilde{\mathbf{Z}}_u \mathbf{P} \right\|_{F_{\mathrm{X}}}^2 = \left\| \mathbf{S} - \widetilde{\mathbf{Z}}_u \widetilde{\mathbf{Z}}_u^\top \mathbf{X} \mathbf{S} \right\|_{F_{\mathrm{X}}}^2 \\
&\text{subject to} \quad \widetilde{\mathbf{Z}}_u^\top \mathbf{X} \widetilde{\mathbf{Z}}_u = \mathbf{I}_{\mathcal{N}}
\end{aligned}
\tag{2.21}
$$

where $\mathbf{I}_{\mathcal{N}} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ is the identity matrix. Once again, the expression of the aforementioned problem is contingent upon the inner product associated with the Hilbert space. Leveraging the fact that the matrix $\boldsymbol{X}$ is tied to a scalar product allows for the formulation of a minimization problem that conveniently omits this matrix, albeit with the requirement of suitably adjusting the snapshot matrix. Indeed, since matrix $\mathbf{X}$ defines a scalar product, it is positive semi-definite. Hence, it possesses a Cholesky decomposition, $\mathbf{X} = \mathbf{X}^{1/2} \left(\mathbf{X}^{1/2}\right)^\top$. As a result, we can rewrite the previous problem, and return to a minimization of the classical Frobenius norm by modifying the snapshot matrices and the resulting basis: $\underline{\mathbf{S}} = \left(\mathbf{X}^{1/2}\right)^\top \mathbf{S}$ and $\underline{\mathbf{Z}}_u = \left(\mathbf{X}^{1/2}\right)^\top \mathbf{Z}_u$. A change of variable reduces to a minimization problem in the sense of the classical Frobenius norm $\|\cdot\|_F$. It illustrates that, when using a specific scalar product, it is feasible to revert to a scalar product associated with a ℓ_2 norm, by appropriately adjusting the snapshots matrices and the bases sought. Thus, the POD basis can be interpreted as a result of the following optimisation problem:

$$
\begin{aligned}
&\underline{\mathbf{Z}}_u = \arg \min_{\widetilde{\mathbf{Z}}_u \in \mathbb{R}^{\mathcal{N} \times \ell}} \quad \left\| \underline{\mathbf{S}} - \widetilde{\mathbf{Z}}_u \widetilde{\mathbf{Z}}_u^\top \underline{\mathbf{S}} \right\|_F^2 := E_{\mathrm{pod},F}\left(\widetilde{\mathbf{Z}}_u, \ell, \underline{\mathbf{S}}\right) \\
&\text{subject to} \quad \widetilde{\mathbf{Z}}_u^\top \widetilde{\mathbf{Z}}_u = \mathbf{I}_{\mathcal{N}}
\end{aligned}
\tag{2.22}
$$

Link with the Singular Value Decomposition

The POD by the method of snapshots relies on the use of the Singular Value Decomposition (SVD) of $\underline{\mathbf{S}}^\top \underline{\mathbf{S}} = \mathbf{S}^\top \mathbf{X} \mathbf{S}$. The SVD decomposition expresses this matrix using two orthogonal matrices $\mathbf{U} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ and $\mathbf{V} \in \mathbb{R}^{n_{\mathrm{train}} \times n_{\mathrm{train}}}$ one diagonal matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{\mathcal{N} \times n_{\mathrm{train}}}$, containing the singular values in descending order. Fixing an a priori order ensures the uniqueness of the decomposition. In the context of the snapshot method, we consider the following correlation matrix:

$$
\underline{\mathbf{S}}^\top \underline{\mathbf{S}} = \left(\mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^\top\right)^\top \left(\mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^\top\right) = \mathbf{V} \boldsymbol{\Sigma}^2 \mathbf{V}^\top
$$

The basis vectors are then reconstructed using the values in the following relationship.

$$
\mathbf{Z}_\ell = \underline{\mathbf{S}} \mathbf{V} \boldsymbol{\Sigma}^{-1} = \underline{\mathbf{S}} \mathbf{V} \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_\ell \end{bmatrix}^{-1}
$$

where σ_1, ..., σ_ℓ are the singular values of $\underline{\mathbf{S}}$ in decreasing order.

Theorem 1 (Schmidt-Eckart-Young) *Let* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *and let* $\mathbf{U}$, $\mathbf{D}$, $\mathbf{V}$ *be the SVD decomposition of* $\mathbf{A}$*, e.g* $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$*. We denote by* r *the ranl of* $\mathbf{A}$*. Let* $\{\boldsymbol{\zeta}_i, \sigma_i^2\}_{i=1}^r$ *(resp.* $\{\boldsymbol{\psi}_i, \sigma_i^2\}_{i=1}^r$*)be the eigenpairs of* $\mathbf{A}\mathbf{A}^\top$ *(resp.* $\mathbf{A}^\top\mathbf{A}$*). Let* $\mathbf{A}_k = \sum\limits_{i=1}^{k} \sigma_i \boldsymbol{\zeta}_i \boldsymbol{\psi}_i^\top$ *with* $k \leq r$*. Then, we have:*

$$\left\|\mathbf{A} - \mathbf{A}_k\right\|_F = \min_{\mathbf{B}\in\mathbb{R}^{m\times n}, rank(\mathbf{B})\leq k} \left\|\mathbf{A} - \mathbf{B}\right\|_F = \sqrt{\sum_{i=k+1}^{r} \sigma_i^2}$$

where $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} A_{i,j}^2}$ *is the Frobenius norm. We also have the alternative definition of* $\mathbf{A}_k = \mathbf{U}_k\mathbf{U}_k^\top\mathbf{A}$*, where* $\mathbf{U}_k = [\boldsymbol{\zeta}_1, ..., \boldsymbol{\zeta}_k]$*, which is equivalent to the other definition.*

The quantity $E_{\text{pod},F}\left(\widetilde{\mathbf{Z}}_u, \ell, \underline{\mathbf{S}}\right)$ defined in Equation (2.22) is referred to as the POD energy. Indeed, thanks to the Schmidt-Eckart-Young theorem, we can link this norm minimization to the sum of the singular values. Namely, we have the following expression of the POD energy:

$$E_{\text{pod},F}\left(\widetilde{\mathbf{Z}}_u, \ell, \underline{\mathbf{S}}\right) = \sum_{i=\ell+1}^{n_{\text{train}}} \sigma_i^2$$

Throughout the previous presentation, we set a base search size constructed by POD, e.g. ℓ base vectors. Interpreting the minimization problem using energy with singular values leads to a slightly different criterion for constructing the reduced basis. Indeed, it means we no longer construct a basis by specifying its size, but by focusing on the amount of relative information it comprises. More precisely, the whole idea is to consider the largest basis, whose POD energy remains below a given threshold. One classical criterion for building the reduced basis by POD is the following: the dimension N_u of the reduced space is typically chosen based on the energy percentage captured by the space:

$$N_u := \min\left\{\ell : \sum_{i=1}^{\ell} \sigma_i^2 \geq \left(1 - \varepsilon_{\text{POD}}^2\right) \sum_{i=1}^{n_{\text{train}}} \sigma_i^2 \right\} \tag{2.23}$$

A swift computation, highlighting the relative information content (RIC) $\sum\limits_{i=1}^{\ell} \sigma_i^2 / \sum\limits_{i=1}^{n_{\text{train}}} \sigma_i^2$, reveals that the criterion on the singular values is equivalent to controlling the energy of POD in relation to the total energy. More precisely, the derivation of this criterion can also be formulated as follows:

$$\sum_{i=1}^{\ell}\sigma_i^2 \geq \left(1-\varepsilon_{\mathrm{POD}}^2\right)\sum_{i=1}^{n_{\mathrm{train}}}\sigma_i^2 \Leftrightarrow \underbrace{\frac{\sum\limits_{i=1}^{\ell}\sigma_i^2}{\sum\limits_{i=1}^{n_{\mathrm{train}}}\sigma_i^2}}_{\mathrm{RIC}} \geq 1-\varepsilon_{\mathrm{POD}}^2 \Leftrightarrow \frac{\sum\limits_{i=1}^{n_{\mathrm{train}}}\sigma_i^2-\sum\limits_{i=1}^{\ell}\sigma_i^2}{\sum\limits_{i=1}^{n_{\mathrm{train}}}\sigma_i^2} \leq \varepsilon_{\mathrm{POD}}^2$$

$$\Leftrightarrow \sum_{i=1}^{n_{\mathrm{train}}}\sigma_i^2-\sum_{i=1}^{\ell}\sigma_i^2 \leq \varepsilon_{\mathrm{POD}}^2\left(\sum_{i=1}^{n_{\mathrm{train}}}\sigma_i^2\right)$$

$$\Leftrightarrow \sum_{i=\ell+1}^{n_{\mathrm{train}}}\sigma_i^2 \leq \varepsilon_{\mathrm{POD}}^2\left(\sum_{i=1}^{n_{\mathrm{train}}}\sigma_i^2\right)$$

$$\Leftrightarrow \sum_{i=\ell+1}^{n_{\mathrm{train}}}\sigma_i^2 \leq \varepsilon_{\mathrm{POD}}^2\left(\sum_{i=1}^{n_{\mathrm{train}}}\sigma_i^2\right)$$

$$\Leftrightarrow \frac{E_{\mathrm{pod},F}\left(\widetilde{\mathbf{Z}}_u,\ell,\underline{\mathbf{S}}\right)}{\sum\limits_{i=1}^{n_{\mathrm{train}}}\sigma_i^2} \leq \varepsilon_{\mathrm{POD}}^2$$

In summary, POD helps us to design an optimal basis in the sense of a scalar product. However, it must be borne in mind that such a basis represents the approximation of a set of previously obtained and computed HF snapshots. This necessary condition entails an incompressible computational cost, as successive calls to the HF solver are needed.

Hands-on implementation of the algorithm

The preceding paragraphs explain the general theoretical concepts behind the POD method. Here, we describe the numerical implementation of this method based on the method of snapshots [Sir87], introducing the formalism for a static case. We assume that we have a training set of discrete snapshots $\{\mathbf{u}_{\mu_j}\}_{j=1}^{n_{\mathrm{train}}}$. The extension to the time-dependent case merely involves adding all the time snapshots for each parameter in the training set. The rest of the methodology remains the same. First, we compute a Gramian $\mathbf{C}_u \in \mathbb{R}^{n_{\mathrm{train}}\times n_{\mathrm{train}}}$, defined as $\left(\mathbf{C}_u\right)_{i,j} = \left(\mathbf{u}_{\mu_i}\right)^{\top}\mathbf{X}\mathbf{u}_{\mu_j}$. In a second step, we solve the eigenvalue problem:

$$\mathbf{C}_u\boldsymbol{\varphi}_n = \lambda_n\boldsymbol{\varphi}_n, \quad \lambda_1 \geq \ldots \geq \lambda_{n_{\mathrm{train}}} \geq 0. \tag{2.24}$$

in order to obtain the eigenpairs $(\lambda_n, \boldsymbol{\varphi}_n)$, for $n = 1, \ldots, N_u$. The number of modes is set by a criterion, for example the one provided before (cf. Equation (2.23)). Once the number of modes is chosen, we can define the POD modes as follows:

$$\boldsymbol{\zeta}_{u,n} = \frac{1}{\sqrt{\lambda_n}}\sum_{j=1}^{n_{\mathrm{train}}}\left(\boldsymbol{\varphi}_n\right)_j\mathbf{u}_{\mu_j} \tag{2.25}$$

In order to introduce compact notation, we shall denote the construction of the reduced basis by POD procedure by the following operator:

$$\mathbf{Z}_u = \mathrm{POD}\left(\{u_{\mu_j}\}_{j=1}^{n_{\mathrm{train}}},\ (\cdot,\cdot)\,,\ \varepsilon_{\mathrm{POD}}\right) \tag{2.26}$$

2.3.4 Data-compression methods within the greedy process

The processes discussed here rely on greedy approaches, involving iterative procedures. At each iteration, new elements $\{\mathbf{u}_{\mu_j}\}_{j=1}^{n_{\text{new}}}$ need to be added to the training set. One approach is to directly include these new elements in the training set and then generate the POD basis using the same operator as before (cf. Equation (2.26)). However, the computational cost associated with constructing the POD basis can become prohibitive if a SVD has to be performed for all parameters at each iteration. Alternatives exist in the literature to implement incremental POD compression approaches, and these are particularly well-suited for the methods explored in this thesis. In the literature, two main classes of methods are identified: hierarchical POD (H-POD [Haa17]) and hierarchical approximate POD (HAPOD [HLR18]). The first method is purely hierarchical, involving the calculation of a POD on the projection of new snapshots onto the reduced space obtained in the previous iteration. This implies that the reduced basis is adapted following the equation below:

$$\mathbf{Z}_u = \left[\mathbf{Z}_u, \mathbf{Z}_{u,\text{new}}\right], \quad \text{with} \quad \mathbf{Z}_{u,\text{new}} = \text{POD}\left(\{\Pi_{\mathcal{Z}^\perp} u_{\mu_j}\}_{j=1}^{n_{\text{new}}},\ (\cdot,\cdot)\,,\ \varepsilon_{\text{POD},u}\right) \tag{2.27}$$

The HAPOD method, on the other hand, does not provide a hierarchical basis. Nevertheless, it still relies on the previously computed reduced basis by incorporating it into the new snapshots (modes multiplied by the square roots of the eigenvalues for consistency):

$$\mathbf{Z}_u = \text{POD}\left(\{\Pi_{\mathcal{Z}^\perp} u_{\mu_j}\}_{j=1}^{n_{\text{new}}} \cup \{\sqrt{\lambda_n}\zeta_{u,n}\}_{n=1}^{N_u},\ (\cdot,\cdot)\,,\ \varepsilon_{\text{POD},u}\right) \tag{2.28}$$

In our work, we rely on the HPOD method in the incremental case to provide a hierarchical basis. Indeed, in terms of data structure, this property facilitates implementation, both in terms of data storage and hyper-reduction processes. Previous works [IST22] have highlighted the challenge of finding an optimal tolerance for the data compression. The compression operator seeks to accurately represent the provided snapshots. However, numerically, the vector projection can never reach absolute zero. Employing POD on these projected snapshots may lead to unreliable results, as the energy content of the projected snapshots could be significantly lower than that of the original snapshot set. This observation explains the importance of introducing a criterion based on the relative projection error. We rely on the regularization approaches [IST22][AAB+22], where the number of modes is chosen according to the following criterion:

$$N^{\text{new}} = \min\left\{M \ : \ \max_{k\in\{1,\ldots,K\}} \frac{\left\|\Pi_{\left(\mathcal{Z}_u \oplus \mathcal{Z}_{u,M}^{\text{new}}\right)^\perp,(.,.)} u_\mu^{\text{hf},(\text{k})}\right\|}{\left\|u_\mu^{\text{hf},(\text{k})}\right\|} \leq \varepsilon_{\text{POD,u}}, \quad \mathcal{Z}_{u,M}^{\text{new}} = \text{span}\left\{\zeta_{u,m}^{\text{new}}\right\}_{m=1}^{M}\right\} \tag{2.29}$$

where $\oplus$ denotes the direct sum of two vector spaces. The reduced basis selectively incorporates basis vectors that effectively minimize the projection error, discarding others treated as noise. Additionally, for numerical efficiency, a criterion is pre-established before computing additional modes. No further POD computation is performed when:

$$\max_{k\in\{1,\ldots,K\}} \frac{\left\|\Pi_{(\mathcal{Z}_u)^\perp,(.,.)} u^{\text{hf},(\text{k})}(\mu)\right\|}{\left\|u_\mu^{\text{hf},(\text{k})}\right\|} \leq \varepsilon_{\text{POD,u}} \tag{2.30}$$

Based on the very same principle, we assume in this situation that the new snapshots belong to the previously generated reduced space, and there is no update of the basis. This preliminary verification avoids unnecessary offline CPU costs.

2.4 Projection-based model order reduction

In the preceding section, our attention was dedicated to constructing the reduced space, that is to say the approximation space for our ROM. Now, our focus shifts to delineating how we evaluate our ROM once this approximation space is established. This presentation unfolds across three aspects: first, the reduced formulation by projection; second, the hyper-reduction process, which addresses the nonlinearities of the problem while retaining computational cost advantages; and finally, the stress field reconstruction process specific to solid mechanics problems when using the chosen hyper-reduction process.

2.4.1 Reduced formulation

The Galerkin ROM is obtained by projecting the discrete residual operator (meaning onto the Equation (2.13)) onto the primal reduced basis $\mathbf{Z}_u$. We first consider the situation without Lagrange multipliers for the implementations of BCs:

$$\mathbf{Z}_u^\top \mathbf{R}_\mu^{\mathrm{hf}} \left(\widehat{\mathbf{u}}_\mu^{(k)},\ \widehat{\mathbf{u}}_\mu^{(k-1)},\ \widehat{\boldsymbol{\mathfrak{S}}}_\mu^{(k-1)} \right) = \mathbf{0}. \tag{2.31}$$

In addressing homogeneous kinematic links (where the right-hand side is null) between the DOFs within the domain Ω in this study, expressed as $\mathbf{B}\mathbf{u}_\mu^{(k)} = \mathbf{0}$, it becomes apparent that specifying these kinematic links as inputs to the online solver becomes unnecessary. Likewise, no specific treatments for these links are required during online resolution. In this approach, the reduced basis inherently accounts for kinematic links, leading to a significant simplification in the coding of the online resolution. It's noteworthy that such a choice can substantially reduce the number of unknowns and, consequently, the computational cost. Indeed, we observe that:

$$\begin{aligned} \mathbf{Z}_u^T \mathbf{R}_\mu^{\mathrm{hf}} \left(\widehat{\mathbf{u}}_\mu^{(k)}, \widehat{\mathbf{u}}_\mu^{(k-1)}, \widehat{\boldsymbol{\mathfrak{S}}}_\mu^{(k-1)} \right) + \mathbf{Z}_u^T \mathbf{B}^T \widehat{\boldsymbol{\lambda}}_\mu^{(k)} &= \mathbf{Z}_u^T \mathbf{R}_\mu^{\mathrm{hf}} \left(\widehat{\mathbf{u}}_\mu^{(k)}, \widehat{\mathbf{u}}_\mu^{(k-1)}, \widehat{\boldsymbol{\mathfrak{S}}}_\mu^{(k-1)} \right) + \left[\mathbf{B}\mathbf{Z}_u \right]^T \widehat{\boldsymbol{\lambda}}_\mu^{(k)} \\ &= \mathbf{Z}_u^T \mathbf{R}_\mu^{\mathrm{hf}} \left(\widehat{\mathbf{u}}_\mu^{(k)}, \widehat{\mathbf{u}}_\mu^{(k-1)}, \widehat{\boldsymbol{\mathfrak{S}}}_\mu^{(k-1)} \right) = \mathbf{0}. \end{aligned} \tag{2.32}$$

By construction, for a given n, the mode $\zeta_{u,n}$ is a linear combination of the snapshots and, therefore, its discrete counterpart should verify $\mathbf{B}\boldsymbol{\zeta}_{u,n} = \mathbf{0}$. This configuration diminishes the count of unknowns since the Lagrange multipliers can be omitted. They are not required in solving the nonlinear system, storing them becomes unnecessary, and they don't play a role in any data compression operation.

2.4.2 Hyper-reduction techniques

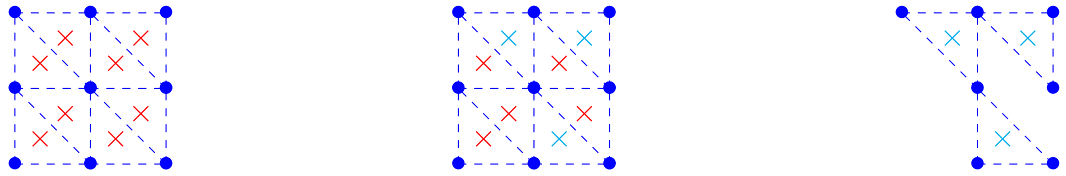
As previously discussed, in the realm of projection-based MOR, hyper-reduction methods are employed to mitigate the online assembling cost associated with variational forms that exhibit nonaffine dependencies on parameters and nonlinearity with respect to the unknown variable. When assembling FE matrices, calculating residuals involves spatial domain integration. The primary objective is to curb the computational complexity of the resulting ROM, which, when scaling with the dimension of the HF discretization, becomes computationally infeasible. Ideally, the goal is to achieve an operation that scales exclusively with the number of modes. This objective matches the ideal case achieved in a linear case. Thus, perfect online efficiency is achieved when the cost of invoking the ROM solver remains independent of the dimension of the full-order model (FOM).

In this thesis, we opt for a hyper-reduction strategy based on empirical quadrature (EQ). Our approach involves constructing a reduced mesh to expedite online assembly costs for the ROM. Here, the term "reduced mesh" refers to a mesh designed by considering a subset of the cells from the HF mesh. More precisely, we focus on the ECSW formulation. This EQ procedure, initially proposed in References [FACC14][FCA15], relies on reweighting either the quadrature points of the mesh or the elemental contributions to approximate the residuals. Several other techniques have been introduced in the literature to overcome the bottleneck induced by the projection step for nonlinear non-affine problems. Additional reweighting methods, such as the Empirical Cubature Method [HCF17], which has been implemented within an industrial context [CA19][CAB+20], have been introduced. Hyper-reduction approaches also include algorithms derived from the Empirical Interpolation Method [BMNP04] family, encompassing its discrete variant, or techniques within the Gappy-POD application, such as A priori Hyper-Reduction [Ryc05], or the Gauss Newton with approximated tensors [CFCA13]. These hyper-reduction methods aim to compute an approximation of the residual form, the cost of which does not scale with the size of the FOM. They can be categorized into two main groups based on their underlying philosophy. The first class seeks an affine-parametric decomposition to approximate the residual form, aligning with the scope of affine-decomposition problems where the RB method demonstrates computational efficiency. Methods falling into this category include the Empirical Interpolation Method and the Discrete Empirical Interpolation Method. The second class operates at the mesh level, aiming to reduce the size of the mesh used to diminish the assembly cost. Techniques within this category encompass the A priori Hyper-Reduction method, energy conserving sampling and weighting approaches, and the empirical quadrature and empirical cubature method. We adopt an EQ approach with ECSW, which has proved its efficiency in numerical solutions for various physical phenomena, namely for solid mechanical problems, both in the static/quasi-static [AAB+22] and dynamical [FACC14][FCA15] framework, for thermo-hydro-mechanical (THM) simulations [IST22], for fluid mechanics [GFTBM21], for magnetostatic [MR22].

The EQ method's philosophy is deeply rooted in the structure of FE codes. We begin by presenting the initially developed method, which relies on all integration points. The hyper-reduction approach aims to establish an effective method for computing the residual form in the reduced problems we intend to assess. The underlying idea is that the evaluation of the residual, and thus the assembly procedure, relies on the computation of successive integrals. All these integrals rely on the application of a common quadrature rule $\boldsymbol{\rho}^{\mathrm{hf}}$. The method seeks to identify the sparsest possible quadrature rule capable of approximating a carefully selected set of integrals (or at the very least, integrals easily accessible without additional computations). This problem can be reformulated as an optimization problem based on signal theory. An alternative approach, the one we opt for in this study, operates at the elemental level. The structure of an FE code is characterized by loops over elements, wherein the total contribution (integral over the entire domain) is computed as a sum of local integrals (integrals over each element). Elemental-level hyper-reduction involves reweighting the elemental contributions to preserve an approximation of the total integral value (over all relevant integrals). The objective is to obtain a new quadrature rule $\boldsymbol{\rho}^{\mathrm{eq}}$ (total or elementary) as sparse as possible, starting from the so-called HF rule $\boldsymbol{\rho}^{\mathrm{hf}}$, which, according to the approach, is tantamount to getting :

$$\left\{\begin{array}{lll} \boldsymbol{\rho}^{\mathrm{hf}} = [\rho_1, \rho_2, ..., \rho_{\mathcal{N}_g-1}, \rho_{\mathcal{N}_g}]^\top \in \mathbb{R}^{\mathcal{N}_g} & \longrightarrow \quad \boldsymbol{\rho}^{\mathrm{eq}} = [0, \rho_2, ..., 0, \rho_{\mathcal{N}_g}]^\top \in \mathbb{R}^{\mathcal{N}_g} & \text{(all integration points)} \\ \boldsymbol{\rho}^{\mathrm{hf}} = [\rho_1, \rho_2, ..., \rho_{N_{\mathrm{e}}-1}, \rho_{N_{\mathrm{e}}}]^\top \in \mathbb{R}^{N_{\mathrm{e}}} & \longrightarrow \quad \boldsymbol{\rho}^{\mathrm{eq}} = [0, \rho_2, ..., 0, \rho_{N_{\mathrm{e}}}]^\top \in \mathbb{R}^{N_{\mathrm{e}}} & \text{(elementwise)} \end{array}\right.$$

There is a natural connection between these two variants of the method. Elemental-level reweighting involves assigning equal weight to all quadrature points. The construction of the reduced mesh then naturally follows (cf. Figure 2.2). During the integration calculations, the elements where all integration weights are zero yield a zero value and can thus be disregarded in the assembly procedure. It is sufficient, therefore, to eliminate all elements with entirely zero quadrature weights in the reduced mesh. In the elemental case, the reduced mesh construction procedure is directly determined by the EQ rule $\boldsymbol{\rho}^{\mathrm{eq}}$ obtained. For comparative results between the two methodologies, element-wise versus point-wise, we refer to the following work [DY22].



(a) $\mathcal{T}_h$ and $\boldsymbol{\rho}^{\mathrm{hf}}$ (b) $\mathcal{T}_h$ and $\boldsymbol{\rho}^{\mathrm{eq}}$ (c) $\mathcal{T}_h^{\mathrm{r}}$ and $\boldsymbol{\rho}^{\mathrm{eq}}$

Figure 2.2: Reduced mesh construction process, starting from the HF mesh and the empirical quadrature rule for a simple mesh. Figure 2.2a depicts an HF mesh. In Figure 2.2b, the red weights are the quadrature weights in $\boldsymbol{\rho}^{\mathrm{eq}}$, which are zero, while the cyan weights are non-zero. This geometry induces the mesh shown in Figure 2.2b.

For simplicity, we exclusively outline the elementwise hyper-reduction procedure for the single-modeling case, which is universally applicable to any nonlinear behavior. The details of extending this method to the multi-modeling scenario will be expounded upon during the analysis of said case (see Chapter 4). As mentioned earlier, the objective is to reweight the elemental contributions to define an empirical residual form $\forall v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$:

$$
\begin{aligned}
\mathcal{R}_\mu^{\mathrm{hf}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{v}\right) &= \sum_{q=1}^{N_e} \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right) \\
&\approx \sum_{q=1}^{N_e} \left(\boldsymbol{\rho}^{\mathrm{eq}}\right)_q \mathcal{R}_{\mu,q}^{\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{u}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right) \\
&:= \mathcal{R}_\mu^{\mathrm{eq}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{v}\right).
\end{aligned}
$$

By taking similar notations as those introduced in Equation (2.7), the objective of the procedure is to provide an empirical residual defined from the empirical quadrature rule $\boldsymbol{\rho}^{\mathrm{eq}}$ as given below $\forall v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$:

$$
\mathcal{R}_\mu^{\mathrm{eq}}\left(\mathbf{u}_\mu^{(k)},\ \mathbf{u}_\mu^{(k-1)},\ \boldsymbol{\sigma}_\mu^{(k-1)},\ \mathbf{v}\right) = \mathcal{R}^{\sigma,\mathrm{eq}}\left(\boldsymbol{\sigma}^{(k)},\ \mathbf{v}\right) = \sum_{q\in I_{\mathrm{eq}}} \left(\boldsymbol{\rho}^{\mathrm{eq}}\right)_q \mathcal{R}_q^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\mathbf{v}\right), \tag{2.33}
$$

where $\{\boldsymbol{\sigma}^{(k)}\}_{k=1}^K$ are the HF stress snapshots of the problem. In the online phase, the solution is sought on the primal reduced space $\mathcal{Z}_{N_u}$. Therefore, it is sufficient to have a good approximation of the residual on the space spanned by the reduced order basis vector $\mathrm{span}\{\boldsymbol{\zeta}_{u,n}\} \subset \mathcal{X}^{\mathrm{hf}}$.

The desired residual form is now at our disposal. The next step involves specifying the conditions that the EQ rule must meet in order to establish a construction protocol. Subsequently, this procedure can be reframed as an optimization problem. Given a tolerance $\delta > 0$, the EQ rule $\boldsymbol{\rho}^{\text{eq}}$ should satisfy the following conditions:

1. the number of nonzero entries in $\boldsymbol{\rho}^{\text{eq}}$ should be as small as possible,
2. the entries of $\boldsymbol{\rho}^{\text{eq}}$ should be non-negative,
3. (*constant-function constraints*) the measure of the domain should be conserved:

$$\left| \sum_{q=1}^{N_e} \rho_q^{\text{eq}} \left| \mathtt{D}_q \right| - |\Omega| \right| < \delta \, |\Omega|$$

4. (*manifold accuracy constraints*) the empirical and HF residuals should be close, meaning that for every primal mode $\boldsymbol{\zeta}_{u,n}$ and HF snapshot $\left(\mathbf{u}^{(k)}, \boldsymbol{\sigma}^{(k)}\right)$, we have:

$$\left| \sum_{q \in I_{\text{eq}}} \rho_q^{\text{eq}} \mathcal{R}_q^{\sigma,\text{hf}} \left(\mathbf{E}_q^{\text{qd}} \boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\text{no}} \boldsymbol{\zeta}_{u,n} \right) - \mathcal{R}^{\sigma,\text{hf}} \left(\boldsymbol{\sigma}^{(k)},\ \boldsymbol{\zeta}_{u,n} \right) \right| \leq \delta \left| \mathcal{R}^{\sigma,\text{hf}} \left(\boldsymbol{\sigma}^{(k)},\ \boldsymbol{\zeta}_{u,n} \right) \right| \tag{2.34}$$

All these constraints enable us to recast the empirical quadrature problem as a ℓ_0 pseudo-norm minimisation problem, known as the sparse representation problem:

$$\min_{\boldsymbol{\rho} \in N_e} \|\boldsymbol{\rho}\|_{\ell_0} \quad \text{s.t.} \quad \left\{ \begin{array}{rcl} \|\mathbf{G}\boldsymbol{\rho} - \mathbf{y}\|_* & \leq & \delta \, \|\mathbf{y}\|_* \\ \boldsymbol{\rho} & \geq & \mathbf{0} \end{array} \right. \tag{2.35}$$

for a suitable choice of $\mathbf{G}$,$\mathbf{y}$, δ and $\|.\|_*$. As analysed in Reference [FACC14], relying on the work [AK98], such a problem is an NP-hard optimization problem.

Attempting to solve such a problem for our hyper-reduction process is therefore illusory. Nevertheless, the minimization problems related to the ℓ_0 pseudo-norm have been extensively explored in the literature on optimization and signal processing. On one hand, established links between the ℓ_1 norm and the ℓ_0 pseudo-norm exist, and the frequent recourse to the latter is aimed at securing sparse solutions. This analytical framework notably underpins the contributions [YP19], where the authors proposed an approximation which relies on the ℓ_1 relaxation of the problem where $\|.\|_* = \|.\|_{\ell_\infty}$. The relaxed problem can thus be reformulated as a linear programming problem, and solved by resorting to appropriate solvers. Alternative relaxation propositions have been advanced in the literature, notably through the utilization of an ℓ_2 norm and the judicious selection of a resolution algorithm capable of endowing the problem with the requisite sparsity. Within this framework, non-negative least squares (NNLS) problems comprise a class of approximation for the sparse representation problem:

$$\boldsymbol{\rho}^{\text{eq}} = \underset{\boldsymbol{\rho} \in \mathbb{R}_+^{N_e}}{\text{argmin}} \|\mathbf{G}\boldsymbol{\rho} - \mathbf{y}\|_2 \tag{2.36}$$

Hyper-reductions methods founded on non-orthogonal matching pursuit algorithms [MZ93][YWD15] have been developped to this end. Those approaches rely on numerical methods for sparse inexact non-negative least-squares initially developped in signal processing. Similarly, Reference FACC14 suggested a methodology called Energy-Conserving Sampling and Weighting method (ECSW) that was built on Lawson and Hanson's algorithm [LH95]. This method constitutes an active-set approach designed to address a non-negative least-squares problem. The algorithm undergoes a key modification through the incorporation of an additional stopping criterion, specifically designed to promote solution sparsity. More precisely, a criterion based on the residuals acquired during the optimization iterations allows for preemptive termination of the iterations:

$$\|\mathbf{G}\boldsymbol{\rho} - \mathbf{y}\|_2 \leq \delta \|\mathbf{y}\|_2 \tag{2.37}$$

where δ is the previously described tolerance (cf. Equation (2.34)). We may compare our implementation approach with previous methods that relied on exact residual operators requiring explicit information about internal variables. Within our work, our strategy is designed to be as non-intrusive as possible, and suitable for an industrial setting. Thus, we choose to reconstruct integrals of the variational formulation outside the HF code, since we wish not to extract the discrete residual from the HF code. Such a strategy involves extracting stress fields, HF gradients of displacement modes, and HF quadrature rules from the HF code, enabling hyper-reduction processes without explicit knowledge of internal variables. The strategy incurs slightly higher memory storage costs and an additional call to the HF code for each calculation of the reduced displacement basis but remains generally applicable, offering a compromise between intrusiveness and efficiency for less general problems. The computation of the empirical quadrature rule by an ECSW approach can be viewed as a call to an operator, which relies on the dictionary $\mathbf{G}$ and a hyper-parameter, more precisely a tolerance δ:

$$\boldsymbol{\rho}^{\mathrm{eq}} = \text{EQ-solve}\left(\mathbf{G},\ \delta\right) \tag{2.38}$$

The various practical steps of the hyper-reduction approach are formalized and illustrated in Figure 2.3. These steps are divided into three stages: dictionary construction, call to optimization solver and reduced mesh construction. The last two steps have been detailed previously, and we now take a moment to elaborate on the first stage. This involves the knowledge of the ROM, namely of the modes $\boldsymbol{\zeta}_{u,i}$, the residual operator $\mathcal{R}_q^{\sigma,\mathrm{hf}}(\cdot,\cdot)$ (in our case, the integral of the double contraction product between stresses and deformations), and stress snapshots $\boldsymbol{\sigma}_\mu^{(k)}$. On Figure 2.3, an example of a dictionary $\mathbf{G}$ is formalized to help the reader understand the type of problem we consider. It's worth noting that alternative strategies could be applied to construct the dictionary, such as considering projections of stress snapshots onto the reduced space or obtaining stress snapshots directly from calling the ROMs. We clarify that the dictionary presented here is not precisely the one used in the practical implementation of our approach to avoid unnecessary complexity in the document. For more details, the reader can refer to Appendix A.3.2.

2.4.3 Reconstruction of the stress by Gappy-POD

At the end of a call to the reduced solver, we acquire the reduced solutions in displacement along with the associated stress derived through the integration of the constitutive equation at sampled elements using empirical quadrature. However, it is noteworthy that these stress vectors do not inherently pertain to the reduced space specifically designated for stress. Instead, these stress vectors are derived from integrating the constitutive law within the HF code, utilizing information derived from the reduced solution in displacement. In our numerical strategy, our approach involves the creation of a second POD basis ($\mathbf{Z}_\sigma$ or $\mathbf{Z}_\mathfrak{S}$, specifically formed using constraint snapshots (or generalized forces). The ultimate goal is to reconstruct a solution constrained within this newly defined reduced space

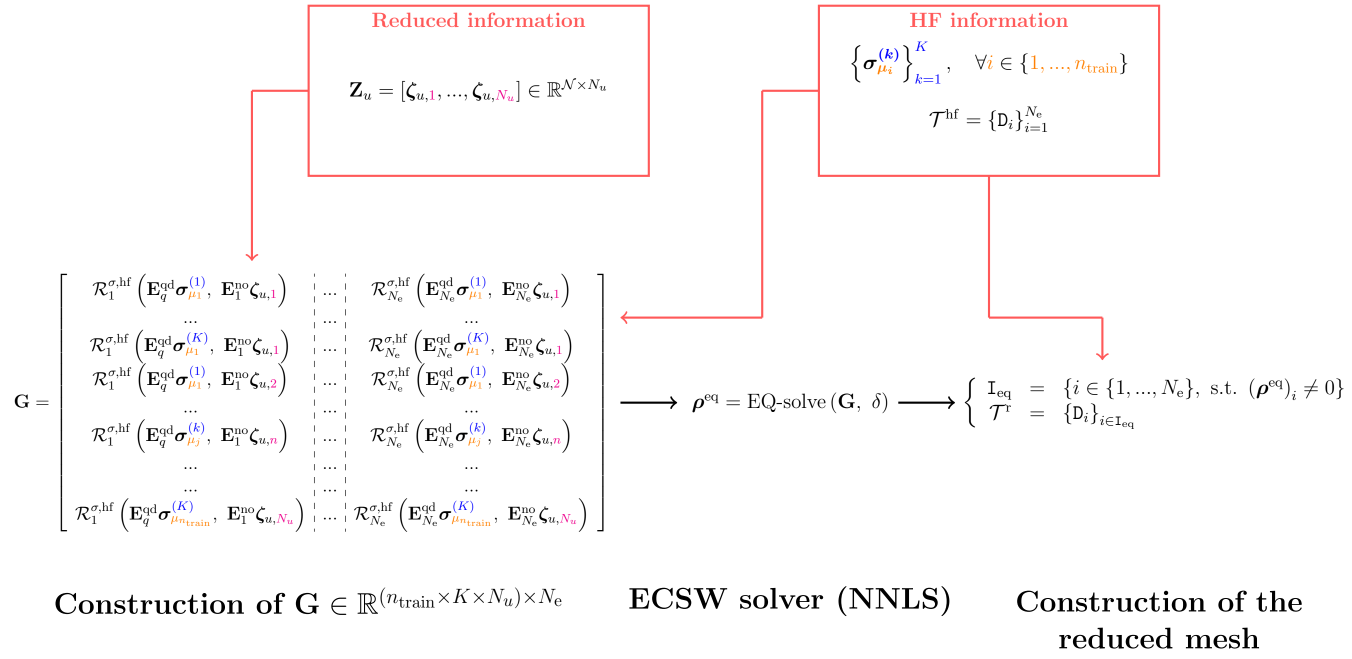


Figure 2.3: Scheme of the steps of the ECSW procedure within our framework: construction of the dictionnary is based on Equation (2.34); the ECSW solver is defined by Equation (2.38); the strategy of construction of the reduced mesh is described on Figure 2.2.

Indeed, when employing reduced meshes, the information regarding stress is confined to the quadrature points of the sampled elements. Yet, a comprehensive understanding of the stress field across the entire HF mesh is essential for an accurate depiction of the mechanical state. Consequently, the reconstruction of the field over the entire mesh becomes imperative. Furthermore, even in the absence of any hyper-reduction procedure, the stresses obtained lack a predetermined association with the earlier generated reduced basis, despite the absence of a reduced mesh. This discrepancy arises from independently generating both reduced bases.

To address these challenges, we implement a Gappy-POD algorithm [ES95] for determining the generalized coordinates. Ultimately, we successfully reconstruct a comprehensive vector based on the knowledge of the vector on the reduced mesh. More specifically, for each time step 'k', we resolve a least-squares problem using the specified mask. In practice, given a stress vector $\boldsymbol{\sigma} \in \mathbb{R}^{\mathcal{N}_g}$ defined at the integration points, we build a vector $\mathbf{n}$ called mask:

$$n_i = \begin{cases} 1, & \text{if } \sigma_i \text{ is related to an integration point on the reduced mesh} \\ 0, & \text{otherwise} \end{cases}$$

The latter is therefore the same size as a full vector, and is used to generate a pointwise multiplication $(\mathbf{n}, \boldsymbol{\sigma})_j = n_j \sigma_j$, $\forall j \in \{1, ..., \mathcal{N}_g\}$. Consequently, the outcome of the operation yields a vector equivalent in size to a full vector (defined on the HF mesh), with components outside the reduced mesh being set to zero. Subsequently, from this resultant product, an associated scalar product and an induced norm can be defined in the following manner:

$$\widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)} = \arg \min_{\boldsymbol{\alpha} \in \mathbb{R}^{N_\sigma}} \left\| \boldsymbol{\sigma}_\mu^{\text{out},(k)} - \mathbf{Z}_\sigma \widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)} \right\|_{\mathbf{n}}^2 \tag{2.39}$$

where the vector $\boldsymbol{\sigma}_\mu^{\text{out},(k)} \in \mathbb{R}^{\mathcal{N}_g}$ is the stress vector designed as follows: the values at the mask are obtained by integrating the constitutive equation (the internal variables are known at these mesh points), and the other values are set to zero.

2.5 *A posteriori* error indicator

2.5.1 General framework

At this point in the presentation, we lack a precise estimate of the discrepancy between the reduced solution and HF solutions (called the approximation error), hence the use of the term *a posteriori* which is calculated once a solution in the ROM is computed. However, there is a crucial imperative to gauge the approximation error in real-time applications for various reasons. Primarily, it is essential to possess the capability to validate a specific level of error incurred by our ROM. Furthermore, the development of an error indicator becomes pivotal as it guides the greedy process for parameter selection in iterative procedures. In this context, we distinguish between the concepts of error estimator and error indicator. While an estimator is used to assess the approximation error, the indicator is solely correlated with the approximation error. Nevertheless, this distinction suffices for constructing a greedy process. To derive such indicators, we choose to rely on residual-based error indicators.

2.5.2 Residual-based error indicator in nonlinear structural mechanics

To define an *a posteriori* indicator, we rely on operator norms constructed from the residual forms appearing in the problem. Given a solution trajectory $\mathbb{U} = \{u^{(k)}\}_{k=0}^{K}$ and the list of time steps of the time-dependent problem $\{\Delta t^{(k)}\}_{k=1}^{K}$, we define the time-discrete $L^2\left(0, t_{\mathrm{f}}\right)$ residual indicator.:

$$\Delta_{N,\mu}^{\mathrm{avg,hf}}\left(\mathbb{U}\right) := \sqrt{\sum_{k=1}^{K} \Delta t^{(k)} \left(\underbrace{\sup_{v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}} \frac{\mathcal{R}_{\mu}^{\mathrm{hf,(k)}}\left(u_{\mu}^{(k)}, u_{\mu}^{(k-1)}, v\right)}{\|v\|_{\mathcal{X}}}}_{:=\Delta_{N,\mu}^{\mathrm{hf,(k)}}} \right)^2}$$

Such a definition is inspired by previous works [HO08] where such *a posteriori* indicators are developed for time-dependent problems. We emphasize that this choice of indicator is not unique. Other works, for instance [IST22], are based on a time-averaged error indicator defined on a time-averaged residual:

$$\sup_{v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}} \frac{\mathcal{R}_{\mu}^{\mathrm{hf,avg}}\left(\mathbb{U}, v\right)}{\|v\|_{\mathcal{X}}}, \quad \text{with} \quad \mathcal{R}_{\mu}^{\mathrm{hf,avg}}\left(\mathbb{U}, v\right) = \sum_{k=1}^{K} \Delta t^{(k)} \mathcal{R}_{\mu}^{\mathrm{hf,(k)}}\left(u_{\mu}^{(k)}, u_{\mu}^{(k-1)}, v\right)$$

In practice, efficiently computing such error indicators is challenging. Indeed, these indicators require calculations based on using HF residuals; thus, involving a total quadrature with a supremum over the entire space $\mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$. Such a computation is not computationally efficient. Consequently, we have developed a strategy consistent with the constructed ROM and the industrial code architecture in order to have cost-efficient error indicators.

Derivation for the single-modeling case

Subsequently, we focus on error indicators for the single-modeling case. We opt for an online/offline strategy that exploits the fact that the stress prediction belongs to a predetermined reduced space $\widehat{\sigma}_\mu^{(k)} \in \operatorname{span}(\zeta_{\sigma,n})$ and that the reduced residual can be expressed accordingly. The objective is to rely on the stress decomposition onto a reduced space to formulate a readily computable error indicator. This imparts a dual purpose to the constraint-reduced basis. In addition to conveying information about the mechanical state throughout the entire mesh, it facilitates the construction of an indicator.

In the following formulation, we assume that the external loadings do not depend on the time variable. This choice is made for the sake of simplicity, and we can refer to Appendix B for more details on the actual procedure when the loading is time-dependent. We introduce the Riesz elements $\psi_n^\sigma \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}}$ associated to the given linear forms:

$$(\psi_n^\sigma,\ v) = \mathcal{L}_n(v), \quad \forall v \in \mathcal{X}_{\mathrm{bc}}^{\mathrm{hf}} \quad \text{with} \left\{ \begin{array}{lcl} \mathcal{L}_n(v) & = & \displaystyle\int_\Omega \zeta_{\sigma,n} : \varepsilon(v)\ dx, \quad 1 \leq n \leq N_\sigma \\[2ex] \mathcal{L}_{N_\sigma+1} & = & \displaystyle\int_\Omega f_v \cdot v\ dx + \int_{\Gamma_n} f_s \cdot v\ ds \end{array} \right. \tag{2.40}$$

By means of the decomposition of the stress solution on the stress reduced basis ($\widehat{\boldsymbol{\sigma}}_\mu^{(k)} = \mathbf{Z}_\sigma \widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)}$), we can restate the residual evaluation with a reduced solution in stress as:

$$\begin{aligned} \mathcal{R}_\mu^{\mathrm{hf,(k)}}\left(\widehat{u}_\mu^{(k)}, \widehat{u}_\mu^{(k-1)}, v\right) \approx \mathcal{R}_\mu^{\sigma,\mathrm{hf}}\left(\widehat{\sigma}_\mu^{(k)}, v\right) &= \mathcal{R}_\mu^{\sigma,\mathrm{hf}}\left(\widehat{\sigma}_\mu^{(k)}, v\right) \\ &= \mathcal{R}_\mu^{\sigma,\mathrm{hf}}\left(\sum_{n=1}^{N_\sigma} \left(\widehat{\alpha}_{\sigma,\mu}^{(k)}\right)_n \zeta_{\sigma,n}, v\right) \\ &= \sum_{n=1}^{N_\sigma} \mathcal{R}_\mu^{\sigma,\mathrm{hf}}\left(\left(\widehat{\alpha}_{\sigma,\mu}^{(k)}\right)_n \zeta_{\sigma,n}, v\right) \end{aligned}$$

Thus, we can recast the dual norm calculation as:

$$\Delta_{N,\mu}^{(k)} = \sup_{v \in \mathcal{X}_{\mathrm{bc}}} \left[\sum_{n=1}^{N_\sigma} \left(\widehat{\alpha}_{\sigma,\mu}^{(k)}\right)_n \frac{\mathcal{L}_n(v)}{\|v\|} - \frac{\mathcal{L}_{N_\sigma+1}(v)}{\|v\|}\right] = \sup_{v \in \mathcal{X}_{\mathrm{bc}}} \frac{\left(\sum_{n=1}^{N_\sigma} \left(\widehat{\alpha}_{\sigma,\mu}^{(k)}\right)_n \psi_n^\sigma - \psi_{N_\sigma+1}^\sigma,\ v\right)}{\|v\|} \tag{2.41}$$

The dual norm is equal to the norm of its Riesz element, which gives a compact expression for the error indicator:

$$\left(\Delta_{N,\mu}^{(k)}\right)^2 = \left\|\sum_{n=1}^{N_\sigma} \left(\widehat{\alpha}_{\sigma,\mu}^{(k)}\right)_n \psi_n^\sigma - \psi_{N_\sigma+1}^\sigma\right\|^2 = \begin{bmatrix} \widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)} \\ -1 \end{bmatrix}^T \boldsymbol{\Sigma}_N \begin{bmatrix} \widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)} \\ -1 \end{bmatrix} = \left(\widetilde{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)}\right)^T \boldsymbol{\Sigma}_N \widetilde{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)} \tag{2.42}$$

where $\boldsymbol{\Sigma}_N \in \mathbb{R}^{N_\sigma+1, N_\sigma+1}$ is the Gramian matrix of the Riesz elements previously introduced, i.e $\left(\boldsymbol{\Sigma}_N\right)_{n,m} = (\psi_n^\sigma,\ \psi_m^\sigma)$, and $\widetilde{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)}$ is the concatenation of the generalized coordinates for the stress with [−1].

Given the absence of numerical validation for the multi-modeling case, a derivation is not explicitly outlined in this thesis. Nonetheless, a derivation for the industrial case is provided in Appendix B.2.

2.5.3 Discussion about implementation constraints in an industrial framework for a single-modeling case

Energy norm choice for the single-modeling case

We shall now discuss the effective calculation of the Riesz elements in the context of the industrial code. However, initially, it is necessary to specify the inner product used, as it determines the Riesz representative. For efficient and code-compatible derivation of an indicator in the single-modeling case, a consistent choice would be to consider the standard H^1 norm: $||u||_{H^1} = \sqrt{\int_\Omega \nabla u \cdot \nabla u + uu \ dx}$. As mentioned previously, our choices are constrained by the industrial context of our implementations. When considering the adoption of such a norm, the finite element code lacks a swift and efficient method for extracting the matrix $\mathbf{X}_u$. To address this challenge, we chose to compress the data using an energy norm. Specifically, we are considering the energy norm associated with a simpler mechanical case, namely linear elasticity. In so doing, we consider the following linear elastic problem (if we omit the time dependence):

$$\left\{\begin{array}{rcll} -\nabla\cdot\sigma_\mu & = & f_v & \text{on} \quad \Omega \\ \sigma_\mu\cdot n & = & f_s & \text{on} \quad \Gamma_n \\ u_\mu & = & 0 & \text{on} \quad \Gamma_d \\ \sigma_\mu & = & \frac{E}{1+\nu}\nabla_s u_\mu + \frac{E}{(1+\nu)(1-2\nu)}\left(\nabla\cdot u_\mu\right)1 & \end{array}\right. \tag{2.43}$$

where E is the Young's modulus and ν is the Poisson coefficient. From a variational point of view, this amounts to considering a case where we are seeking a displacement field $u \in \mathcal{X}_{\text{bc}}^{\text{hf}}$ such that :

$$a_\mu\left(u,v\right) = F(v) \quad \text{with} \quad \left\{\begin{array}{rcl} a_\mu\left(u,v\right) & = & \displaystyle\int_\Omega \frac{E}{1+\nu}\nabla_s u : \nabla_s v + \frac{E}{\left(1+\nu\right)\left(1-2\nu\right)}\left(\nabla\cdot u\right)\left(\nabla\cdot v\right)\ dx \\ F(v) & = & \displaystyle\int_\Omega f_v v + \int_{\Gamma_n} f_s v \end{array}\right. \tag{2.44}$$

The bilinear form $a_\mu : \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ is symmetric, coercive and continuous. As a consequence of Korn and Poincaré's inequalities, it defines an equivalent norm of H^1: $\forall w \in \mathcal{X}, \ \|w\|_{a_\mu} = \sqrt{a_\mu\left(u,v\right)}$. However, this energy norm is parametric. We shall have a parameter-independent norm, and thus, we choose rely on the energy norm for the centroid of the parameters $\overline{\mu} \in \mathcal{P}$: $\mathbf{X}_u = \mathbf{K}_{\overline{\mu}}$, where $\mathbf{K}_{\overline{\mu}}$ is the stiffness matrix obtained for an elastic problem and the vector of parameters $\overline{\mu}$ (or at the components of the vector corresponding to the elastic behaviour).

Derivation of the Riesz representatives

We can now shed light on the actual computation of the Riesz representatives. In a general manner, these vectors can be determined by solving $N_\sigma + 1$ linear systems defined by Equation (2.40) and will hence fulfil the boundary conditions associated with the system: $\mathbf{B}\boldsymbol{\psi}_n^\sigma = \mathbf{0}$. In our framework, it is not straightforward to formulate a problem in variational form by hand or to extract all information to solve Equation (2.40) algebraically outside the FE solver. Nevertheless, functionalities exist to extract Riesz elements of the given linear forms but on a larger space $\mathcal{X}^{\text{hf}}$, i.e. for vectors that do not satisfy the problem BCs. Indeed, such features are often implemented in industrial-grade FE codes so that engineers can have access to internal forces vectors or support reaction forces. Such vectors are often used by engineers and defined as follows:

$$\forall \mathbf{v} \in \mathbb{R}^{\mathcal{N}}, \qquad \left\{ \begin{array}{rcl} (\mathbf{F}_n,\ \mathbf{v})_{\ell_2} & = & \int_\Omega \zeta_{\sigma,n} : \varepsilon(v)\ dx, \qquad \forall n \in \{1, ..., N_\sigma\} \\ (\mathbf{F}_{N_\sigma+1},\ \mathbf{v})_{\ell_2} & = & \int_\Omega f_v \cdot v\ dx + \int_{\Gamma_n} f_s \cdot v\ ds \end{array} \right. \tag{2.45}$$

As a reminder, the Riesz elements should belong to the same space as the displacement space. Therefore, as mentioned previously, the scalar product used for their definition is the scalar product associated to the energy norm for the parameter centroid. One may see from Equation (2.40) that ψ_n^σ is solution to a quadratic optimization problem associated with cost function $\mathbf{v} \mapsto \frac{1}{2}\mathbf{v}^T\mathbf{K}_{\overline{\mu}}\mathbf{v} - \mathbf{v}^T\mathbf{F}_n$ under the equality constraint $\mathbf{B}\mathbf{v} = \mathbf{0}$. The Karush-Kuhn-Tucker (KKT) optimality conditions read:

$$\left\{ \begin{array}{rcl} \mathbf{K}_{\overline{\mu}}\boldsymbol{\psi}_n^\sigma + \mathbf{B}^T\boldsymbol{\lambda} & = & \mathbf{F}_n, \\ \mathbf{B}\boldsymbol{\psi}_n^\sigma & = & \mathbf{0} \end{array} \right. \tag{2.46}$$

Actually, Equation (2.46) defines an easy problem to provide as an input to a FE solver: it is a linear elastic case for the parameter centroid, with the very same BCs as the HF problem, and an explicit field of nodal forces as an external load (previously computed by Equation (2.45)) It is therefore sufficient to use the HF solver for $N_\sigma + 1$ linear problems. Finally, the parameter-independent matrix that appears in the error indicator definition is computed as follows:

$$(\boldsymbol{\Sigma}_N)_{n,m} = (\psi_n^\sigma,\ \psi_m^\sigma)_{a_{\overline{\mu}}} = \boldsymbol{\psi}_n^\sigma \cdot (\mathbf{K}_{\overline{\mu}} \cdot \boldsymbol{\psi}_m^\sigma) = (\boldsymbol{\psi}_n^\sigma)^T\, \mathbf{K}_{\overline{\mu}} \cdot \boldsymbol{\psi}_m^\sigma, \qquad \forall n, m \in \{1, \cdots, N_\sigma + 1\} \tag{2.47}$$

For a given HF FE solver, this strategy is a non-intrusive way to compute the error indicator, since there is no need to retrieve $\mathbf{B}$ or $\mathbf{K}_{\overline{\mu}}$ matrices from the FE solver.

2.6 Overview of the reduction methodology

To comprehensively elucidate the methodology employed in this thesis, we offer an overview of the approach. This step back from the detailed mathematical implementation which has been thoroughly expounded in the preceding sections, allows for a broader perspective on the algorithms' overarching structure before delving into subsequent chapters focused on numerical results. The presentation of the methodology unfolds in two stages. Initially, we define our approach without incorporating the greedy aspect, that is, by considering the pre-given sampling of parameters or, for instance, by considering a non-parametric scenario. Such a scenario, often termed the solution reproduction problem, constitutes a necessary step in testing a model reduction approach. This case involves replicating a HF problem with a reduced one. Addressing this sub-problem facilitates the design of algorithmic blocks that can be readily repurposed within the parametric framework, as well as the provision of validation tests for the latter. In the second stage, we present a comprehensive overview of the algorithm, taking the greedy aspect into account for sampling the parametric space.

2.6.1 Construction of the reduced order model within the offline/online paradigm

Without accounting for the parameter sampling process, we begin by encapsulating the reduction methodology employed. A comprehensive delineation is provided within the offline/online paradigm. Initially, leveraging precomputed HF snapshots, we orchestrate the construction of the reduced model. This entails formulating a condensed basis for displacements $\mathbf{Z}_u$, pivotal for subsequent projection, an empirical quadrature scheme $\boldsymbol{\rho}^{\mathrm{eq}}$, and the derived reduced mesh $\mathcal{T}_h^{\mathrm{r}}$, essentials for CPU computational gain. Furthermore, a basis for constraints $\mathbf{Z}_\sigma$ or generalized force $\mathbf{Z}_{\mathfrak{S}}$ becomes mandatory for reconstructing the mechanical state of the structure on the HF mesh. All numerical methodologies used are described earlier and collectively constitute the offline phase of the algorithm. The ensuing online phase entails the evaluation of this ROM. In practical terms, the computational cost of this evaluation, namely the reduced problem resolution, must unequivocally be cheaper than its HF counterpart. In our scenario, the evaluation occurs in two steps. Initially, a projection-driven resolution yields coordinates of reduced displacements. The displacement solution is then fully known. However, in practice, to determine the material's mechanical state, hence its stress state, it is necessary to perform Gappy-POD to obtain it. Thus, the online phase produces two sets of generalized coordinates. The summary of the required computation steps can be found in the Algorithm 1.

Algorithm 1 Key steps of the construction of a ROM: offline/online decomposition

```
Offline step
    Compute the HF-snapshots                                        ▷ Call of HF code:code_aster
    Construction of the reduced order basis ($\mathcal{Z}_u$ and $\mathcal{Z}_\sigma$)       ▷ Section 2.3.3 and 2.3.4
    Empirical Quadrature procedure $\boldsymbol{\rho}^{\mathrm{eq}}$                      ▷ Section 2.4.2
Online step
    Compute the primal generalized coordinates $\left\{\widehat{\boldsymbol{\alpha}}_{u,\mu}^{(k)}\right\}_{k=1}^{K}$          ▷ Equation (2.31)
    Compute the dual generalized coordinates using Gappy-POD $\left\{\widehat{\boldsymbol{\alpha}}_{\sigma,\mu}^{(k)}\right\}_{k=1}^{K}$    ▷ Section 2.4.3
```

2.6.2 POD-Greedy sampling relying on ECSW procedure

We have designed a POD-Greedy procedure to systematically explore the parametric space for constructing the reduced model. For the single-modeling scenario, we have devised a weak-greedy approach, while for multi-modeling, a strong-greedy methodology has been developed, coupled with a tailored implementation compatible with our industrial code. Supplementary remarks regarding the extension of the weak-greedy approach to multi-modeling are provided in Appendix B.2. In both instances, mesh hyper-reduction is accomplished through an ECSW approach applied to three-dimensional elements (and also surface elements: additional details on implementations are elucidated in the Appendix). The iterative construction of the reduced model continues until a user-defined criterion is met. Figure 2.4 offers a graphical representation of the algorithm employed. The distinctive features of both cases are highlighted using a color-coded scheme, with augmentations specific to multi-modeling depicted in red.

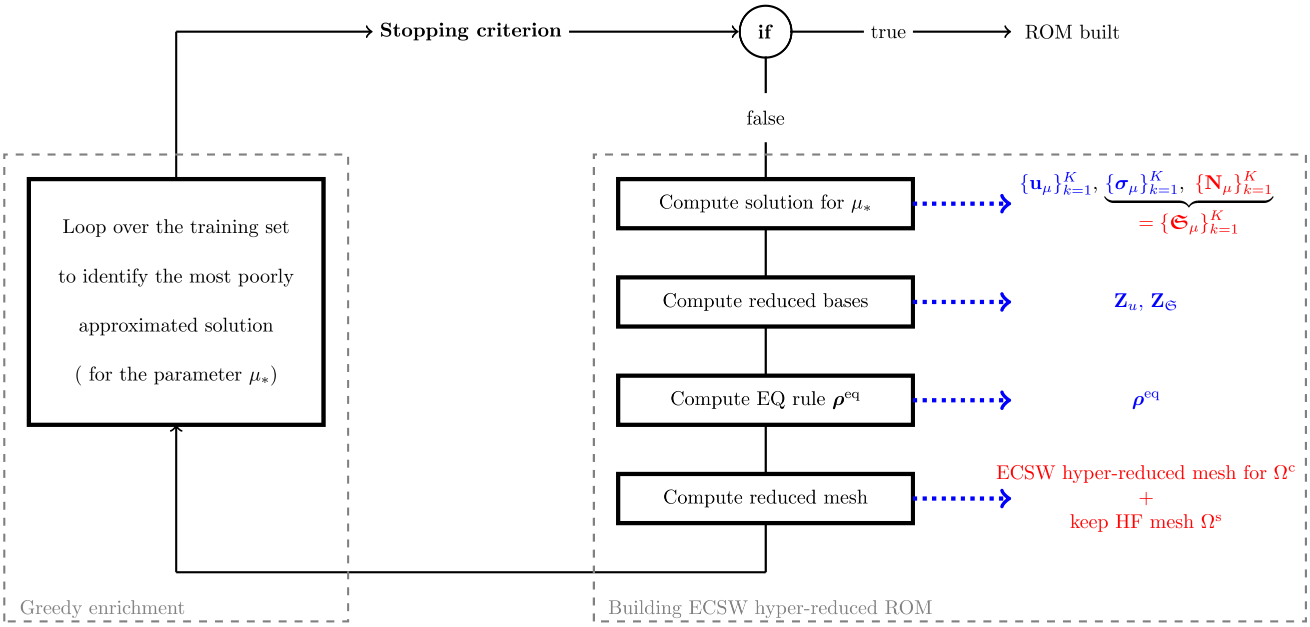


Figure 2.4: Schematic diagram of the greedy process implemented to build a reduced model in the context of single-modeling and multi-modeling problems.

Chapter 3

Model problems relying on an industrial-grade finite element code for structural mechanics

3.1 Introduction

The purpose of this section is to present the problem formulations investigated in this thesis. Specifically, two types of modeling are discussed within the scope of this work: first, the single-modeling paradigm, scrutinized through the lens of an elastoplastic holed-plate subjected to tension; and second, and more importantly, the intricate industrial scenario encapsulating a standard section of a NCB.

The focal point of our modeling aspirations is the section of the NCB, a test case laden with intricate mechanical phenomena instigated by thermal and hydric influences. More specifically, we focus on a block of concrete, with a nonlinear rheological behavior. Notably, this case unfolds as a multi-modeling challenge, given that the containment structures in nuclear facilities are made of with prestressed concrete. This material is represented in EDF's engineering studies by kinematic coupling between two materials, concrete and steel. In summary, our overarching objective is to construct a ROM capable of a time-dependent problem, potentially activated by weak coupling phenomena and featuring multi-modeling intricacies, all implemented in an industrially relevant FE code (`code_aster`).

Within this framework, the implementation of a three-dimensional single-modeling case can be seen as an initial validation step for our proposed methodology. From a numerical standpoint, this affords us the opportunity to scrutinize diverse methodologies, encompassing hyper-reduction and projection-based solvers, within an industrial code for a nonlinear three-dimensional problem, with temporal evolution (or at least a pseudo-time in the case of static analysis). The chosen problem for this test is a three-dimensional elastoplastic plate with a hole, a classical solid mechanics test case. Details regarding the model choices and geometry are provided in Section 3.2.

We dedicate additional effort in Section 3.3 to elaborate on the physical phenomena at play and the modeling approach used for the industrial case of a standard section of a NCB. First, we explain the physics of the problem, followed by the development of governing equations to model such behavior. The choice of governing equations is not the focus of this research and is based on accumulated knowledge within EDF's R&D. Finally, we present the specific geometry studied in this thesis, namely, a representative volume of a containment building section. Again, the selection of this geometry and applied conditions stems from engineering studies to develop a model and corresponds to the three-dimensional model used in practice for leakage studies. We conclude by providing some numerical examples to enhance understanding of the underlying physical phenomena at play.

The theoretical and numerical elements presented in this section are derived from online preprints, specifically from the following works:

- ◇ [AAB$^+$24b] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. International Journal for Numerical Methods in Engineering, 125(4):e7385, 2024.,
- ◇ [AAB$^+$24a] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, Sylvie Michel-Ponnelle, and Tommaso Taddei. Projection-based model order reduction for prestressed concrete with an application to the standard section of a nuclear containment building. *arXiv preprint arXiv:2401.05098*, 2024.

3.2 Single-modeling approach: application to an elastoplastic analysis of a plate with a hole

We previously introduced a methodology to address the case of single-modeling for handling three-dimensional materials with nonlinear behaviors and internal variables. As this approach can be seen as a step towards our complex model, we chose to define an intermediate validation case to test our approach. This case must be consistent with the introduced notations and feature nonlinearity that does not introduce additional challenges to our problem. We opted for the case of a holed plate in elasto-plasticity. This geometry and behavior constitute a classic and straightforward validation case in solid mechanics. The geometry is relatively simple, which eases analysis and modeling. Furthermore, the elasto-plastic behavior introduces a common nonlinearity, as plastic deformations often occur in real situations where the material reaches its elastic limit. Additionally, the interpretation of results is facilitated by this type of behavior, as the interaction between elastic and plastic regimes is clear to interpret for simple numerical models. Moreover, the numerical treatment of an elasto-plastic behavior allows both static and time-dependent tests, providing a clear validation of the algorithm's steps. Hence, this scenario serves as an excellent option for the initial validation of our reduction methodology..

We validate our reduction procedure through the examination of a three-dimensional elasto-plastic holed-plate submitted to tensile loading. In this section, we begin by elucidating the physical formulation of the material's constitutive equations (Section 3.2.1). Then, we delve into the resolution algorithm employed in our study (Section 3.2.2), followed by a detailed explanation of the configuration used in our numerical illustration (Section 3.2.3). To enhance clarity, we may eliminate the parametric dependence from the notation in this section, symbolizing the omission of the subscript μ.

3.2.1 Continuous equations

To ensure that the terminology is clear and understandable for all readers, we shall briefly reintroduce some of the concepts required to define the behavior laws used in our work. Details of mechanical principles are not the main focus of our work, and the reader is invited to refer to the relevant literature if further or more detailed information is required. We provide some examples for elasticity [Sal05][LT06] or elasto-plasticity [HS87][BCCF01].

Plasticity in materials mechanics refers to the proficiency of a material to undergo permanent deformation after being subjected to stress beyond its elastic limit. In simple terms, when a material becomes plastic, it can undergo deformation that persists even after the mechanical load is removed. Prior to reaching the elastic limit, the material experiences reversible elastic deformation, returning to its initial shape when the stress is removed. During this period, and for a specific range of loads, the material remains in an elastic regime. Beyond the elastic limit, the material enters the plastic domain and undergoes permanent deformation. The material enters a regime of irreversible phenomena. Generally, plasticity can be characterized by the relationship between stress (force per unit area) and plastic deformation (or derived quantities), depicted by a hardening curve. Hardening in materials mechanics occurs when a material becomes stiffer and stronger as it undergoes plastic deformations. A classic example of hardening is isotropic hardening, where resistance to plastic deformation increases equally in all directions of the material. In other words, the material hardness uniformly increases, irrespective of the direction in which plastic deformation occurs.

Within this work, we consider a small-displacement small-strain mechanical problem. We assume that the total deformation is the sum of a plastic part (ε^{p}) and an elastic part ($\varepsilon^{\mathrm{el}}$):

$$\varepsilon = \varepsilon^{\mathrm{el}} + \varepsilon^{\mathrm{p}}$$

where the plastic deformation comprises the irreversible part of the behavior. The elastic behavior depends on two parameters, the Young's modulus E and the Poisson coefficient ν. The elastic constitutive equation is :

$$\sigma = \mathcal{F}^{\sigma}\left(\nabla_s u, \varepsilon^p\right) = \frac{E\nu}{(1+\nu)(1-2\nu)}\mathrm{Tr}\left(\nabla_s u - \varepsilon^p\right)1 + \frac{E}{1+\nu}\left(e - e^p\right) \tag{3.1}$$

where the deviator of the strain and stress tensors are introduced:

$$e = \mathrm{dev}\left(\nabla_s u\right), \quad e^p = \mathrm{dev}(\varepsilon^p), \quad s = \mathrm{dev}(\sigma), \qquad \text{where} \qquad \mathrm{dev}(\tau) = \tau - \frac{1}{3}\mathrm{Tr}(\tau)1 \tag{3.2}$$

As a reminder, these tensor quantities are employed to describe the mechanical behavior of a material. For instance, the deviator of strains excludes the volumetric component (or uniform dilation/contraction) of the strain tensor. It focuses on local deformations that alter the shape of the material without changing its total volume. The deviator of strains and stresses serves to isolate the non-uniform, local, and deviatoric components of the total tensor quantities.

In this work, we consider a Von Mises criterion for an isotropic hardening. The Von Mises criterion is based on representing the stress state in terms of an equivalent strain invariant. This invariant is calculated from the principal components of stresses in a given state and is used for comparison with a material-specific critical value. The mathematical expression of the Von Mises criterion for a material in three dimensions is often formulated as follows:

$$\sqrt{\frac{3}{2}s : s} \leq \sigma_y$$

where σ_y is the initial elastic limit of the material. If the inequality above is satisfied, the material is considered to be in an elastic state. If violated, indicating that the deviatoric stress exceeds the threshold stress, the material undergoes plastic deformation.

We deviate momentarily in our model presentation to highlight that this category of problem falls perfectly within the scope of the Equation (2.1). Indeed, in our analysis, the internal variables (γ) that appear in the model are the plastic strain (ε^p) and the cumulative plastic strain (p). In the framework of the formulations presented in the previous chapter, this decision implies that the evolution equations on the internal variables are expressed using the following system:

$$
(\dot{\varepsilon}^p, \dot{p}) = \mathcal{F}^\gamma(\sigma, \varepsilon^p, p) \quad \Leftrightarrow \left\{ \begin{array}{lr} \sigma^{\text{eq}} = \sqrt{\frac{3}{2} s : s} & \\ \sigma^{\text{eq}} - R(p) \leq 0 & \text{[Von Mises criterion]} \\ p(t) = \sqrt{\frac{3}{2}} \int_0^t \left\| \dot{\varepsilon}^p(\tau) \right\| \, d\tau & \\ \dot{\varepsilon}^p = \dot{p} \frac{3}{2\sigma^{\text{eq}}} s \qquad \dot{p} \geq 0 \quad \dot{p}\left[\sigma^{\text{eq}} - R(p)\right] = 0 & \text{[Normality rule]} \end{array} \right. \tag{3.3}
$$

where σ^{eq} is an Von Mises equivalent stress and $R(p)$ denotes the elastic limit, and evolves as a function of the cumulative plastic strain p. Details regarding the definition of the hardening curve for our numerical cases are presented later (see Section 3.2.3).

3.2.2 Incremental algorithm for elastoplastic solvers

We provide here the choice of the time discretization algorithm used to solve the elasto-plastic problem. The time integration of the mechanical behavior of the problem is performed from the computation of a deformation increment:

$$
\varepsilon^{(k)} = \varepsilon^{(k-1)} + \Delta\varepsilon^{(k-1)} \tag{3.4}
$$

This formulation is consistent with the assumption of one-setp time integrators. We recall that e (resp. s) stands for the deviator of the strain (resp. stress) tensor. The discretization of the problem results in finding $\left(\Delta p^{(k-1)},\ \Delta\varepsilon^{p,(k-1)}\right)$ for a given $\Delta\varepsilon^{(k-1)}$ such that:

$$
\left\{ \begin{array}{lcl} p^{(k)} & = & p^{(k-1)} + \Delta p^{(k-1)} \\ \varepsilon^{p,(k)} & = & \varepsilon^{p,(k-1)} + \Delta\varepsilon^{p,(k-1)} \end{array} \right. \tag{3.5}
$$

with:

$$
\left\{ \begin{array}{rcl} \sigma^{(k)} & = & \sigma^{(k-1)} + \dfrac{E\nu}{(1+\nu)(1-2\nu)} \operatorname{Tr}\left(\Delta\varepsilon^{(k)}\right) + \dfrac{E}{1+\nu}\left(\Delta e^{(k)} - \Delta\varepsilon^{p,(k)}\right) \\ \sigma^{\text{eq},(k)} & - & R\left(p^{(k-1)} + \Delta p^{(k-1)}\right) \leq 0 \\ \Delta\varepsilon^{p,(k-1)} & = & \Delta p^{(k-1)} \dfrac{3}{2\sigma^{\text{eq},(k)}} s^{(k)} \\ \Delta p^{(k)} & \geq & 0 \\ \Delta p^{(k)} & & \left[\sigma^{\text{eq},(k)} - R\left(p^{(k-1)} + \Delta p^{(k-1)}\right)\right] = 0 \end{array} \right. \tag{3.6}
$$

We choose to consider an algorithm referred to as incremental in the literature, with a first-order accurate time discretization. The solution varies depending on whether the evolution is exclusively elastic or elastoplastic. Such a procedure adopted is referred to as the return mapping algorithm (or radial return)[W$^+$63]. It resorts to an elastic prediction phase, where the stress field is derived under the assumption of a purely elastic material ($\sigma_{n+1}^{\text{elas}}$). The function $f\left(\sigma_{n+1}^{\text{elas}},\ p_n\right)$ is then estimated based on this prediction. If the solution obtained remains in the elastic region, the next iteration can be launched. Otherwise, a correction is performed by solving the nonlinear equation:

$$f\left(\sigma_{n+1}, p_n + \Delta p_n\right) = 0 \tag{3.7}$$

This equation is nonlinear and is solved through a Newton solver (secant method). The set of unknowns is inferred from the plastic deformation increment Δp_n. Note that this algorithm is even applied for static problems. In this case, a pseudo-time is introduced. From a physical perspective, it can be understood as a time modeling the evolution of the irreversibility within the material.

Algorithm 2 Return mapping algorithm

```
Computation of the elastic prediction $s_{n+1}^{\text{elas}} = s_n + 2\mu\Delta e_n$
Stress computation $\sigma_{n+1}^{\text{elas}},\ \sigma_{n+1}^{\text{elas,eq}}$
Computation of the criterium $f\left(\sigma_{n+1}^{\text{elas}},\ p_n\right)$
if $f\left(\sigma_{n+1}^{\text{elas}},\ p_n\right) \leq 0$ then                                ▷ Elastic Evolution
    Computation of the stress and internal variables:
        $\sigma_{n+1} = \sigma_{n+1}^{\text{elas}}, \quad \varepsilon_{n+1}^{p} = \varepsilon_{n}^{p}, \quad p_{n+1} = p_n$
end if
if $f\left(\sigma_{n+1}^{\text{elas}},\ p_n\right) > 0$ then                                   ▷ Elastoplastic
    Find $\Delta p_n$ solution of                                                   ▷ Equation (3.7)
        $\sigma_{n+1}^{\text{elas,eq}} - \frac{3E}{2(1+\nu)}\Delta p_n - R\left(p_n + \Delta p_n\right) = 0$
    Computation of plastic deformation increments:
        $\varepsilon_{n+1}^{p} = \varepsilon_{n+}^{p} + \Delta\varepsilon_{n}^{p}, \qquad p_{n+1} = p_n + \Delta p_n$
    Update of stress and internal variable:                                         ▷ Equation (3.1)
        $\sigma_{n+1} = \sigma_n + \frac{E\nu}{(1+\nu)(1-2\nu)}\mathrm{Tr}\left(\Delta\varepsilon_n\right)1 + \frac{E}{1+\nu}\left(\Delta e_n - \Delta e_n^{p}\right)$
end if
```

3.2.3 Choice of the hardening curve

The work hardening curve is chosen to follow a power law (referred as `VMIS_ISOT_PUIS` in the `code_aster` database), which implies that the elastic limit evolves on the accumulated plastic strain as follows:

$$R(p) = \sigma_y + \sigma_y \left(\frac{E}{a_{\text{pui}} \sigma_y} p \right)^{\frac{1}{n_{\text{pui}}}}$$

where n, a_{pui} are strain hardening coefficients and σ_y is the initial elastic limit. This algorithm provides us with stable responses on a range of parameters. The resolution procedure used in this work is a elastic predictor-return mapping (plastic corrector)[W$^+$63]. In case of a plastic evolution, the nonlinear equation that ensures the fulfillment of the criterion is solved using the secant method. All the physical parameters of the problem are summarized in Table 3.1.

E	ν	σ_y	n_{pui}	a_{pui}
MPa	no dim.	MPa	no dim.	no dim.

Table 3.1: Summary of the physical parameters.

3.2.4 Holed plate under tensile loading

Geometric configuration and BCs

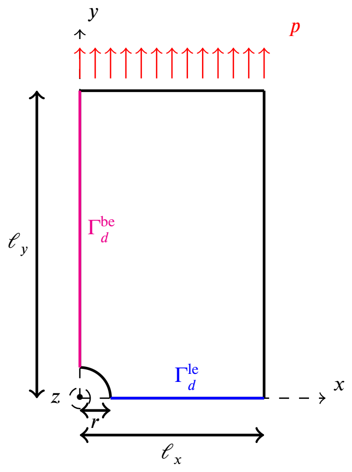


Figure 3.1: Geometric configuration and loading for the elasto-plastic plate with a hole.

As mentioned earlier, we shall study the problem of a three-dimensional plate with a circular hole in its centre and subjected to a tension force. The geometrical domain is narrowed for reasons of symmetry (geometry given on Figure (3.1)). We consider that the tension force is only applied on the upper boundary of the plate. We assume that the vertical displacement is homogeneous on the upper boundary, where symmetric boundaries are also applied:

$$\begin{cases} -\nabla \cdot \sigma = f_v & \text{on } \Omega \\ \sigma \cdot e_y = -p & \text{on } \Gamma_n^{\text{up}} \end{cases}, \quad \text{such that} \quad \begin{cases} u_y = 0 & \text{on} \quad \Gamma_d^{\text{be}} \\ u_x = 0 & \text{on} \quad \Gamma_d^{\text{le}} \\ u_z = 0 & \text{on} \quad \Gamma_d^{\text{ba}} \\ u_z(x_1) \;=\; u_z(x_2), & \forall (x_1, x_2) \in \Gamma_n^{\text{up}} \times \Gamma_n^{\text{up}} \end{cases} \tag{3.8}$$

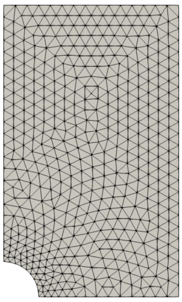
where the associated boundaries are defined as:

$$\begin{cases} \Gamma_d^{\mathrm{be}} &= \{y=0,\ \forall(x,z)\in[r,\ell_x]\times[0,\ell_z]\} \\ \Gamma_d^{\mathrm{le}} &= \{y=0,\ \forall(y,z)\in[r,\ell_y]\times[0,\ell_z]\} \\ \Gamma_d^{\mathrm{ba}} &= \{y=0,\ \forall(x,y)\in[r,\ell_x]\times[0,\ell_z]\setminus\{(x,y),\quad \text{s.t. } x^2+y^2<r\}\} \\ \Gamma_n^{\mathrm{up}} &= \{(x,\ell_y,z),\ \forall x\,[0,\ell_x]\,,\forall z\in[0,\ell_z]\} \end{cases}$$

We mainly consider the case of a constant applied force, but we also allow for variable applied pressure.

Finite element discretization

We consider a three-dimensional quadratic tetrahedral mesh for our numerical investigations. We provide the mesh information in the Table 3.2b, and a visualization of the mesh (cf. Figure 3.2a).



$\mathcal{N}_e$	$\mathcal{N}^{\mathrm{no}}$	$\mathcal{N}^{\mathrm{qd}}$	$\mathcal{N}$	$\mathcal{N}_g$
11 981	18 446	59 905	55 338	359 430

(a) Mesh visualization.

(b) Mesh information.

Figure 3.2: Details on theesh used for the numerical example for a single-modeling material: mesh vizualisation (cf. Figure 3.2a); mesh information (cf. Figure 3.2b) : number of three-dimensional cells ($\mathcal{N}_e$), number of nodes ($\mathcal{N}^{\mathrm{no}}$), number of three-dimensional quadrature points ($\mathcal{N}^{\mathrm{qd}}$), size of the discretized displacement ($\mathcal{N}$) and stress vectors ($\mathcal{N}_g$).

3.3 Thermo-Hydro-Mechanical (THM) modeling of large concrete structures

In this section, we start with a detailed presentation of the physical phenomena we are studying, more specifically the Thermo-Hydro-Mechanical (THM) study for large prestressed concrete structures. First, we present the crucial aspects of concrete mechanics, in order to justify and understand the chosen numerical modeling. Next, we introduce the numerical model used. We also detail the assumptions underlying our model, highlighting the choices made to faithfully represent the mechanisms at play. This second step should also show how this type of problem fits into the more general framework we have adopted. Finally, we turn to the practical application of our methodology by presenting the specific problem on which we are testing our model. This includes specific assumptions on geometry, boundary conditions and a predefined mesh. In particular, our approach aims to speed up actual calculations by using a pre-existing model from EDF's engineering and R&D departments, considered here as the HF problem we seek to reproduce.

3.3.1 Phenomena related to the aging of concrete in large-scale structures.

Concrete, as a complex, porous material, exhibits dynamic behavior influenced by its heterogeneous nature and the temporal evolution of the properties of its components. This complexity stems from its composition, a subtle blend of aggregates, cement, water, and any mineral additions and admixtures, carefully proportioned to meet mechanical and durability requirements. Cement, the key ingredient in concrete, is produced by firing limestone and clay at high temperatures, resulting in clinker which, when mixed with water, triggers a chemical process called hydration [Bye99][T$^+$97]. Hydration is a crucial stage in the hardening of concrete. When water is added to cement, chemical reactions occur between the cement's components, forming calcium hydrate crystals. This process generates heat, known as the heat of hydration, which contributes to the evolution of the concrete's mechanical properties. At the same time, concrete undergoes a solidification phase, during which the cement paste changes from a plastic to a solid state. This initial hardening provides the necessary strength for concrete to support structural loads. However, complete hardening may take several weeks or even months, during which time the concrete may be vulnerable to temporary deformations, notably endogenous shrinkage and desiccation shrinkage. Thermo-hydro phenomena within concrete encompass interactions between heat generated by hydration, external temperature variations, and the presence of water. These interactions can influence the physical properties of concrete, such as thermal shrinkage and crack development.

Shrinkage

In order to understand the phenomena involved, we briefly present some of the mechanical phenomena that induce deformation in concrete. Shrinkage refers to the expansion of volume that occurs during or after the setting of concrete. This phenomenon can be induced by the hydration of the concrete, as well as by variations in temperature or the water content of the concrete. To comprehend various forms of shrinkage, it can be beneficial to categorize contributions based on the conditions and interactions with the surrounding environment. Endogenous conditions for concrete refer to an environment where the concrete is subject to internal influences, with no significant exchange of water with the outside. In other words, concrete is isolated from the external environment in terms of water supply or significant water losses. Under such conditions, various types of shrinkage can occur in concrete. Plastic shrinkage is a contraction that occurs in concrete before it hardens. It arises from a combination of chemical shrinkage, linked to hydration reactions, and desiccation shrinkage, associated with water evaporation from the concrete surface. The hydration reaction, also referred to as Le Châtelier contraction [LC04], involves the formation of hydrates with a volume less than the sum of anhydrous cement and water volumes. Thermal shrinkage occurs due to the material cooling after internal heating caused by exothermic hydration reactions. As time progresses, water in the capillary pores is consumed by the cement's anhydrides' hydration, termed self-drying, without external water exchange.

Under external conditions, including thermo-hydric loads, additional deformations are associated with desiccation shrinkage. Desiccation in the context of concrete refers to the water loss from the material, typically due to the evaporation of water from its surface. This occurrence stems from the interplay between concrete and the external environment, particularly when external conditions promote rapid water evaporation from the concrete surface. Specifically, this phenomenon arises from the water imbalance between the evaporated water at the concrete surface and that retained within its mass, leading to material contraction. This physical phenomenum induces transport mechanisms, highly dependent on the ambient relative humidity (RH)[XBMJ94][Che11]. RH is defined as the ratio of vapor pressure to saturation vapor pressure for a given temperature. For large structures, these phenomena mainly pertain to the hydric behavior of the outer layer of concrete rather than the average hydric behavior across the structure's thickness [BBB$^+$18]. Desiccation shrinkage is, therefore, a concrete contraction resulting from desiccation, specifically the loss of water through evaporation.

Creep

Creep refers to the gradual, slow deformation of a material subjected to a constant stress over time. In the context of concrete, creep can lead to additional deformations beyond the initial hardening phase, and it becomes crucial to account for these effects in the design and long-term performance evaluation of concrete structures. While hypotheses about creep mechanisms may vary, there is universal acknowledgment of the crucial role of water, leading to the identification of two creep modes in the short and long term [Ben02]. The diverse responses of concrete to different stresses constitute a vital area of study for comprehending its behavior under various conditions. Basic creep and desiccation creep are two separate phenomena that impact the performance of concrete under specific circumstances. Basic creep pertains to the prolonged deformation of a concrete structure enduring constant loading, primarily induced by stress relaxation within the structure over time. The creep process can extend for a considerable duration, and the resultant deformation is typically irreversible. In contrast, desiccation creep is a deformation resulting from the loss of water from the concrete.

Prestressed concrete

Prestressed concrete is a technique employed in civil engineering to enhance the strength and durability of concrete structures. The principle revolves around introducing prestress, i.e. applying internal forces before the application of service loads. Steel cables, also known as tendons, are placed in the concrete before it sets, then stretched and anchored. As the concrete hardens, these tendons exert a compressive force on the material, thereby enhancing its tensile strength. The purpose is to counteract the tensile forces anticipated in the structure. Thus, such a method helps reduce deformations, maximize load-carrying capacity. Since such a technique is widely used in civil engineering to optimize the design of structures and enhance their ability to withstand significant loads over extended periods, it has also been used for large containment structures on which we focus. The installation of a prestressed concrete structure involves tensioning the cables within the concrete. In our case, the tension profile along a cable adheres to an official standard (BPEL 91 regulation [Règ92]). Considering physical parameters, such as initial tension, a tension profile is computed along the cable's length relative to the curvilinear abscissa. The coupling between cables and concrete can be delineated into three primary stages: before, during, and after prestressing. In the initial stages of these structures, concrete is poured around sheaths and initiates the drying process. Subsequently, cables are inserted into these ducts and prestressed to meet civil engineering standards. Finally, cement is poured into the ducts, facilitating the continuation of the structure's life with a kinematic coupling between the concrete and the steel cables.

3.3.2 Weak-coupling strategy for the THM numerical model

In this section, we introduce the mathematical model designed to simulate the behavior of prestressed concrete. We consider models that account for the evolution of large concrete structures over their lifetime, which consists mainly of two stages: young age and long-term evolution. The young age refers to a stage during which the chemical and physical properties of concrete are changing at a fast rate, as it sets and hardens. Long-term phase represents the evolution of hardened concrete under operating conditions (taking into account thermo-hydric loadings) and mechanical loadings. Within the framework of the FE models employed in practice, we only consider long-term evolution. Indeed, as stated in the introduction, the motivation for this work lies in the study of concrete ageing. To this end, real data are measured over periods of several years, and numerical simulations must be able to cope with the long-term simulation of these structures. Furthermore, from a purely experimental perspective, engineers possess very limited data on young age, as the measurement tools available are not set up. This is another reason why the focus of numerical approaches is primarily on long-term studies.

As indicated above, concrete behavior results from complex physicochemical phenomena, involving multiphysical couplings. Indeed, the phenomena that occur over the lifetime of concrete are intimately linked to the presence of water in the material and its evolution, as well as to the temperature of the material. Such material behavior requires a THM modeling strategy: the behavior of the material is based on knowledge of the temperature (T), the water content (C_w) in the concrete and the mechanical fields, in a framework where all these phenomena are coupled together. Since we are interested in modeling the whole ageing of the concrete structure, our THM model should encompass the various physical processes which induce deformations within the concrete: shrinkage, dessication and creep.

Notation	**Physical quantity**	**Unit**
T	Temperature	K
ξ	Hydration degree	–
h	Ambient relative humidity (RH)	–
C_w	Water content of concrete	–
σ	Stress in the concrete	Pa
$\varepsilon^{\mathrm{c}} = \nabla_s u^{\mathrm{c}}$	Deformations in the concrete	–
N	Normal efforts in the prestressing cables	N
$\varepsilon^{\mathrm{s}} = \partial_s u^{\mathrm{s}}$	Deformations in the prestressing cables	–

Table 3.2: Fields of interest in the overall THM model for large prestressed concrete structures.

In our framework, we adopt a weakly-coupled approach. This assumption implies that the computation is carried out in a chained manner. First, a thermal calculation is performed, followed by a hydric calculation (water diffusion in the concrete). Once all the thermal and hydric fields are known, a mechanical calculation is conducted. Each calculation step yields fields of interest which: first describe the state of the material; second, can be reused for subsequent calculation steps. At the end of the thermal calculation, we get the temperature field (T) and the degree of hydration of the concrete (ξ; which will be always analytically given in our simulations); at the end of the hydric calculation, we get the water content of concrete (C_w); at the end of the mechanical calculation, we get the displacement fields in the steel cables and in the concrete, the associated deformations ($\varepsilon = [\varepsilon^{\mathrm{c}}, \varepsilon^{\mathrm{s}}]$), the stresses in the concrete (σ) and the normal forces in the cables (N). Table 3.2 details the ouputs for the entire THM calculation. The different steps in the process are summarized in Figure 3.3. Such a formulation of the problem is founded on several assumptions. To begin with, the influence of the mechanical response on the thermal and water fields is neglected [JPCGH07][BBB+18]. Furthermore, it is assumed that the hydric response has no influence on the thermal fields. Weak-coupling approaches have demonstrated their effectiveness in modeling prestressed concrete structures, both for a Representative Structural Volume (RSV) [JPCGH07][BBB+18] and for a full-scale model [ACMC16].

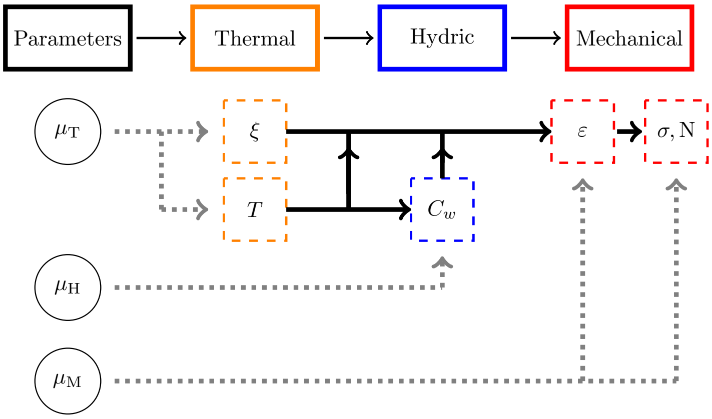


Figure 3.3: Weakly-coupled chained THM approach for large prestressed concrete structures.

3.3.3 THM constitutive equations

As stated above, we describe in the following section the set of equations that make up the THM problem under study. Prestressed concrete behavior modeling requires a multi-modeling approach: a three-dimensional nonlinear rheological model is used for concrete; and prestressing cables are described by a one-dimensional linear thermo-elastic behavior. As mentioned above, the rheological behavior of concrete is coupled with hydric and thermal phenomena. Thermal-hydric resolutions are thus solved on the concrete domain (Ω^{c}), while mechanical calculations are solved on both domains (Ω^{c} and Ω^{s}).

In the following paragraphs, we present the various constitutive equations solved to determine the mechanical state of the problem. We start with equations related to thermal and hydric behaviors, solved at the level of concrete. Next, we highlight the constitutive equations for the rheological model of concrete, followed by the one-dimensional model applied to the cables. We also elaborate on the coupling between these different domains.

Modeling of the thermal and the hydric behavior of the concrete

First, we introduce the set of equations employed for the first two stages of the chained calculation: the thermal calculation and the hydric calculation. This decision is motivated by the fact that this calculation is the starting point for the mechanical calculation, to which we apply our model reduction methodology. The temperature evolution is modeled by the heat equation [Fou88]:

$$\rho_c c_p^p \frac{\partial T}{\partial t} = \nabla \cdot (\lambda_c \nabla T), \quad \text{on } \Omega_c, \tag{3.9}$$

where ρ_c is the density of the concrete, c_p^p heat capacity of hardened concrete and λ_c thermal conductivity of hardened concrete. Dirichlet conditions are applied in our context (see details for the numerical test case in Section 3.3.4).

Since we only consider liquid water diffusion [Gra95], moisture transfer is modeled by a single nonlinear diffusion on C_w (cf. Equation (3.10a)), which denotes the water content of the concrete. The diffusion equation depends on D_w, which is a phenomenological diffusion coefficient, and is assumed to follow Arrhenius' law [BN72]. In summary, the nonlinear diffusion equation of the water content can be summed up as follows:

$$\begin{cases} \dfrac{\partial C_w}{\partial t} = \nabla \cdot \left[D_w \left(C_w,\ T \right) \nabla C_w \right], \quad \text{on } \Omega_\text{c}, & (3.10\text{a}) \\ D_w \left(C_w,\ T \right) = D_{w,0} \left(C_w \right) \dfrac{T}{T_w^0} \exp \left[-\dfrac{U_w}{R} \left(\dfrac{1}{T} - \dfrac{1}{T_w^0} \right) \right], & (3.10\text{b}) \\ D_{w,0} \left(C_w \right) = A \exp \left(B C_w \right), & (3.10\text{c}) \end{cases}$$

where U_w is the activation energy of drying, R the ideal gas constant and $D_{w,0}\left(C_w\right)$ is the diffusion coefficient at a reference temerature T_w^0. The latter is assumed to follow a model defined by previous work [MAA88], which depends on two model parameters A, and B.

At the scale of large concrete structures, measurements of ambient conditions cannot be made in terms of the water content of the concrete, and are thus conducted in relative humidity [Bou16]. For the sake of consistency and use of collected data, the boundary conditions are formulated in terms of RH. From an experimental point of view, the drying or wetting cycles are assumed to affect only the concrete skin. This assumption enables to draw a link between the water concentration in the concrete and the relative humidity. For a given temperature, these two quantities are related by a bijective function called the sorption-desorption function:

$$C_w = f_d \left(h \right). \tag{3.11}$$

Within the framework of these constitutive laws, the sorption-desorption function may be defined either analytically with hyper-parameters [BBB+18][VG80], or empirically by defining a function. In our case, we define a sorption-desorption function as shown in Figure 3.4. This curve is drawn from experimentally acquired points (without interpolation).

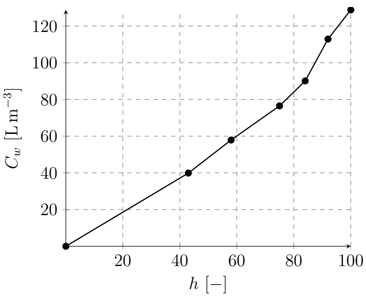
When using real life data, the BC of the water diffusion problem are stated in terms of RH. All the parameters related to the thermal and hydric aspects of the model are summarized and detailed in Table 3.3.

Modeling of the mechanical behavior of the concrete

We now detail the governing equations for the mechanical behavior of concrete. Since, we consider small-displacement small-strain mechanical problems, the total strain is decomposed as the sum of several contributions:

$$\varepsilon = \varepsilon^{\text{el}} + \varepsilon^{\text{th}} + \varepsilon^{\text{ds}} + \varepsilon^{\text{bc}} + \varepsilon^{\text{dc}},$$

where ε^{el} is the elastic strain tensor, ε^{th} the thermal strain tensor, ε^{ds} the dessication shrinkage strain tensor, ε^{bc} the basic creep strain tensor, ε^{dc} the dessication creep strain tensor and ε^{en} the autogenous shrinkage. We explain in the following section the evolution and constitutive equation



(a) Sorption-desorption function used for the THM problem.

C_w [L m^{-3}]	0	39.0	57.9	76.5	90.1	112.9	128.8
h [−]	0	43	58	75	84	92	100

(b) Summary.

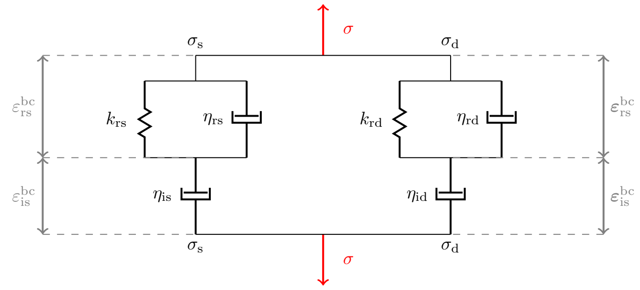
Figure 3.4: Definition of the sorption-desorption function f_d (defined in Equation (3.11)). The table shows the point values given to define the function. The function is computed by linear interpolation between those points. The reference configuration corresponds to $h = 100$, which is the initial RH value in the wall.

Calculation step	**Notation**	**Physical quantity or parameter**	**Unit**
Thermal (μ_{T})	ρ_c	Density	kg m^{-3}
	c_c^p	Heat capacity of hardened concrete	kJ kg K^{-1}
	λ_c	Thermal conductivity of hardened concrete	W m^{-1} K^{-1}
Hydric (μ_{H})	D_w	Phenomenological diffusion coefficient	
	f_d	Sorption-desorption function	
	T_w^0	Reference temperature	K
	$D_{w,0}$	Diffusion coefficient at a reference temperature T_w^0	
	U_w	Activation energy of drying	kJ mol^{-1}
	R	Ideal gas constant	kJ mol^{-1} K^{-1}
	A	Model parameter for Mensi's law	10^{-15}m^2 s^{-1}
	B	Model parameter for Mensi's law	−

Table 3.3: Summary of parameters and physical quantities at stake in the modeling of the thermal (cf. Equation (3.9) and the hydric (cf. Equation (3.10a)-(3.10b)-(3.10c)) behavior.

that help expressing the different strain tensors.

According to experimental observations, the variation of thermal strain $\varepsilon^{\mathrm{th}}$ is proportionnal to temperature variations (cf. Equation (3.12a)). The proportionality coefficient $\alpha_{\mathrm{th,c}}$ is referred to as the thermal dilation coefficient of concrete and is assumed to be constant when focusing on the long-terme phase. Similar experimental observations suggest a linear dependency between the variations of the dessication shrinkage strains $\varepsilon^{\mathrm{ds}}$ and the water content of the concrete C_w (cf. Equation (3.12b)), which is expressed thanks to dessication shrinkage coefficient (α_{ds}). We assume that we have the same kind of relationship between the autogenous shrinkage $\varepsilon^{\mathrm{en}}$ and the



(a) Burger rheological model for the basic creep.

Notation	Physical quantity or parameter	Unit
E_{c}	Young's modulus (concrete)	Pa
ν_{c}	Poisson's ratio (concrete)	–
$\alpha_{\mathrm{th,c}}$	Thermal dilation coefficient (concrete)	K^{-1}
α_{dc}	Dessication shrinkage coefficient	–
β_{endo}	Autogeneous shrinkage coefficient	–
ν_{bc}	Basic creep Poisson's ratio	–
k_{rd}	Reversible deviatoric basic stiffness	Pa
η_{rd}	Reversible deviatoric basic viscosity	Pa s
η_{id}	Irreversible deviatoric basic viscosity	Pa s
U_{bc}	Basic creep activation energy	$\mathrm{kJ\,mol}^{-1}$
T_{bc}^{0}	Basic creep reference temperature	°C
κ	Basic creep consolidation parameter	–
η_{dc}	Desiccation creep parameter	Pa s

(b) Summary of the parameters for the mechanical model.

Figure 3.5: Parameters for the three-dimensional mechanical model (concrete).

hydratation degree ξ, expressed thanks to a β_{endo} coefficient:

$$\begin{cases} \dot{\varepsilon}^{\mathrm{th}} = \alpha_{\mathrm{th,c}} \dfrac{\partial T}{\partial t} \mathrm{I}, & (3.12a) \\ \dot{\varepsilon}^{\mathrm{ds}} = \alpha_{\mathrm{dc}} \dfrac{\partial C_w}{\partial t} \mathrm{I}, & (3.12b) \\ \dot{\varepsilon}^{\mathrm{en}} = \beta_{\mathrm{endo}} \dfrac{\partial \xi}{\partial t} \mathrm{I}. & (3.12c) \end{cases}$$

The model selected for the creep deformations is the Burger model developed by previous work [FMPG12]. This choice is motivated by several experimental validations and is well-suited for creep investigations on the considered structures, as confirmed by previous work [BBB+18]. We assume that the creep is a phenomenon involving a decoupling of a spherical part and a deviatoric part. We decompose the Cauchy stress tensor (σ) as the sum of a spherical part (σ_{s}) and deviatoric part (σ_{d}):

$$\sigma = \sigma_{\mathrm{s}} \mathrm{I} + \sigma_{\mathrm{d}},$$

where $\sigma_{\mathrm{s}} = \mathrm{Tr}(\sigma)/3$, and I is the identity tensor. The Burger creep model is built on a decomposition into a reversible and an irreversible part, where we split each tensor into its spherical and deviatoric part:

$$\begin{cases} \varepsilon &= \varepsilon_{\mathrm{i}}^{\mathrm{bc}} + \varepsilon_{\mathrm{r}}^{\mathrm{bc}}, \\ \varepsilon_{\mathrm{i}}^{\mathrm{bc}} &= \varepsilon_{\mathrm{rs}}^{\mathrm{bc}} \mathrm{I} + \varepsilon_{\mathrm{rd}}^{\mathrm{bc}}, \\ \varepsilon_{r}^{\mathrm{bc}} &= \varepsilon_{\mathrm{is}}^{\mathrm{bc}} \mathrm{I} + \varepsilon_{\mathrm{id}}^{\mathrm{bc}}. \end{cases}$$

Each part (deviatoric and spherical) is described by a Burger-type model. For each chain, the reversible basic creep strains are modeled by a Kelvin-Voigt rheological elements, whereas the irreversible basic creep strains are modeled by Maxwell elements. The Kelvin-Voigt model (cf. Equation (3.13a)) used for the reversible reversible spherical basic creep is expressed thanks to the stiffness (resp. viscosity) k_{rs} (resp. η_{rs}), while the irreversible spherical basic creep viscosity η_{is} is given by a nonlinear relationship, expressed thanks to a consolidation parameter κ (cf. Equation

(3.13b)).

$$\begin{cases} h\sigma_{\mathrm{s}} = k_{\mathrm{rs}}\varepsilon_{\mathrm{rs}}^{\mathrm{bc}} + \eta_{\mathrm{rs}}\dot{\varepsilon}_{\mathrm{rs}}^{\mathrm{bc}}, & (3.13\mathrm{a}) \\ h\sigma_{\mathrm{s}} = \underbrace{\eta_{\mathrm{is}}^{0} \exp\left(\dfrac{\left\|\varepsilon_{i}^{\mathrm{bc}}\right\|_{m}}{\kappa}\right)}_{:=\eta_{\mathrm{is}}} \dot{\varepsilon}_{\mathrm{is}}^{\mathrm{bc}}, \text{ where } \left\|\varepsilon_{i}^{\mathrm{bc}}\right\|_{m} = \max_{\tau\in[0,t]} \sqrt{\varepsilon_{i}^{\mathrm{bc}}(\tau) : \varepsilon_{i}^{\mathrm{bc}}(\tau)}, \quad \forall t \geq 0, & (3.13\mathrm{b}) \end{cases}$$

The deviatoric part is expressed using a similar set of tensor equations (the spherical part being a set of scalar equations). The aforementioned model accounts for thermo-activation of basic creep. To this end, stiffness and viscosity parameters expressions follow an Arrhenius' law:

$$\kappa(T) = \kappa_0 \exp\left[-\frac{U_{\mathrm{bc}}}{R}\left(\frac{1}{T} - \frac{1}{T_{bc}^{0}}\right)\right],$$

where k_{rs}^{0} is the reversible spherical creep stiffness at a reference temperature T_{bc}^{0} and U_{bc} the activation energy of basic creep. Finally, the equivalence of spherical and deviatoric chains enables to restrict the number of model parameters, by assuming a constant creep Poisson ratio ν_{bc}, given by the following relation:

$$\frac{k_{\mathrm{rs}}}{k_{\mathrm{rd}}} = \frac{\eta_{\mathrm{rs}}}{\eta_{\mathrm{rd}}} = \frac{\eta_{\mathrm{rs}}^{0}}{\eta_{\mathrm{rd}}^{0}} = \frac{1 + \nu_{\mathrm{bc}}}{1 - 2\nu_{\mathrm{bc}}}.$$

In order to model the dessication creep strain, we consider the following equation, founded on the work of previous work [BC85]:

$$\dot{\varepsilon}^{\mathrm{bc}} = \frac{1}{\eta_{\mathrm{dc}}}\left|\dot{h}\right|\sigma,$$

where η_{dc} is a material parameter (Pa s).

Modeling of the coupling between concrete and prestressing cables

To echo our underlying premises, we have two models, one corresponding to concrete, which we have just presented, and a second corresponding to steel, which we will detail below. In the numerical model studied here, the one-dimensional mesh (modeling the steel cables) is immersed within the three-dimensional mesh. This means that the cables "cross" the concrete cells. A kinematic linkage is performed in order to connect the concrete nodes and the steel nodes. Since the coupling is assumed to be perfect (no slip between the tendons and the cement), coincident points in each material are assumed to have the same displacement. Instantaneous prestressing losses due to anchor recoil and friction are not taken into account at the scale of the considered RSV.

Furthermore, the cables are modeled by bars, which means that we resort to a one-dimensional approach where only the tension-compression forces are considered. In this framework, the structure is described at each instant by a curve representing its mean line. Consequently, only the normal efforts appear (efforts defined along the tangent vector to the beam section) in the variational formulation of the problem. Two sets of equilibrium equations appear in the studied case: during the prestressing step (namely between the times $t^{\mathrm{init,p}}$ and $t^{\mathrm{end,p}}$) and after the prestressing step (namely until the end of the study t_{f}):

$$\begin{cases} \partial_s \mathrm{N}\left(s,\ t\right) &= f_\mathrm{s}, \quad \forall t \in \left[t^{\mathrm{init,p}}\ , t^{\mathrm{end,p}}\right], \quad \text{and} \quad [\![\mathrm{N}]\!]\left(x_i^{\mathrm{no,1d}}\right) = -\frac{t^{(k)}-t^{\mathrm{init,p}}}{t^{\mathrm{end,p}}-t^{\mathrm{init,p}}}F_i, \\ \partial_s \mathrm{N}\left(s,\ t\right) &= f_\mathrm{s}, \quad \forall t \in \left[t^{\mathrm{end,p}}, t_\mathrm{f}\right], \end{cases} \tag{3.14}$$

where $\{x_i^{\mathrm{no,1d}}\}_{i=1}^{\mathcal{N}^s}$ are the nodes of the one-dimensional mesh and F_i are the nodal forces prescribed in order to respect the BPEL regulation used. We consider a linear thermo-elastic constitutive equation for the steel cables. Thus, the normal efforts in the cables (N) are linked to the uniaxial strains (ε^s) in the cables:

$$\mathrm{N} = E_s S_s \left(\varepsilon^\mathrm{s} - \alpha_{\mathrm{th,s}} \Delta T\right),$$

where E_s the Young's modulus, $\alpha_{\mathrm{th,s}}$ the thermal dilation coefficient, S_s the section of the prestressing cables and ΔT is the temperature rise in the beam. Details and informations of the physical parameters for the three-dimensional are provided on Figure 3.3, whereas those on the one-dimensional are given on Figure 3.6.

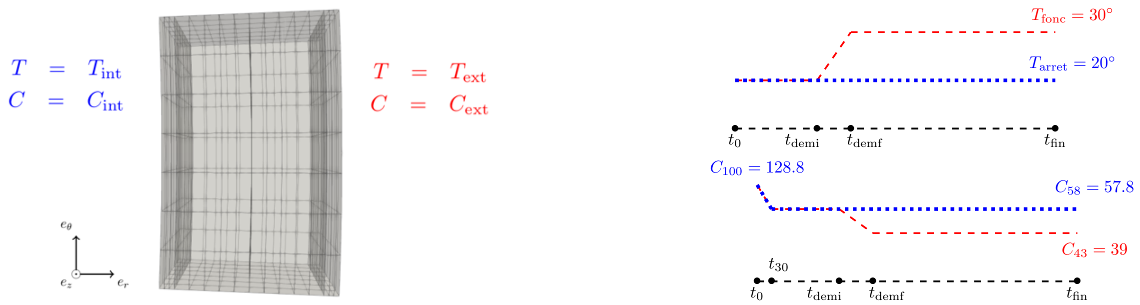
Notation	**Physical quantity or parameter**	**Unit**
E_s	Young's modulus (steel)	Pa
ν_s	Poisson's ratio (steel)	–
ρ_s	Density (steel)	$\mathrm{kg\,m^3}$
$\alpha_{\mathrm{th,s}}$	Thermal dilation coefficient (steel)	$\mathrm{K^{-1}}$
S_s	Cable cross-sections	$\mathrm{m^2}$

Figure 3.6: Parameters for the one-dimensional mechanical model (steel).

3.3.4 Representative Structural Volume : standard section of a nuclear containment building

The physical model is designed to capture the behavior of the so-called standard zone of the model, which corresponds to a portion of the mesh at mid-height, in the cylindrical part of the NCB. Thus, the region covered by the Representative Structural Volume (RSV) comprises a three-dimensional portion containing three tangential prestressing cables and two vertical cables. For the section studied in this study, the internal radius of the wall is 21.9m, while the external radius is 23.4m. The width of the standard section corresponds to an angular sector of 4.2. For the scope of this work, the effect of passive steel reinforcement is being neglected.

Two mesh designs are used in practice: one for thermo-hydric calculations and another for mechanical calculations. The thermal mesh is refined close to the intrados and extrados to enable better reconstruction of the thermo-hydric gradients. The fields resulting from this procedure are then projected onto the mechanical mesh. We highlight the fact that the meshes employed in these studies are fairly coarse. In fact, these meshes have been built in order to be able to carry out uncertainty quantification or data assimilation studies. Therefore, engineers had to strike a balance between affordable computational cost and approximation quality. Numerical solutions for thermal problems may exhibit oscillations (in terms of temporal and spatial discretizations). This may imply a violation of the maximum principle. To avoid this phenomenon, linear finite

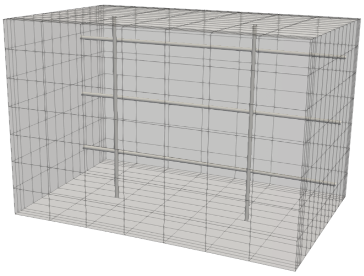


(a) Temperature and water content BCs.

(b) Temperature and water content evolutions.

Figure 3.7: BCs for the thermal and hydric problems visualized on the HF thermal mesh.

elements and a lumping of the mass matrix are used for this study. As previously mentioned, the thermal mesh does not contain the prestressing cables: it is composed of linear hexahedral cells (`HEXA8`). For the mechanical mesh, hexahedral quadratic elements (`HEXA20`) are employed for the concrete, and prestressing tendons are represented using `SEG2` linear finite elements (2-node beams).



(a) Vizualisation of the mechanical mesh.

N_{e}	N_{e}^{1d}	N_{e}^{2d}	N_{e}^{3d}	$\mathcal{N}$	$\mathcal{N}_{\text{c}}$	$\mathcal{N}_{\text{s}}$
1532	784	693	55	4076	3911	165

(b) Summary of the parameters for the one-dimensional mechanical model.

Figure 3.8: Visualization of the mechanical mesh (cf. Figure 3.8a) and information on the mechanical mesh (number of elements and number of nodes for one- and three-dimensional meshes, Figure 3.8b).

The BCs and loads applied to the RSV zone are detailed below (cf. Equation (3.15)). Figure 3.7 shows the temperature and water content histories adopted for the thermo-hydraulic calculations. As mentioned above, the BCs applied are Dirichlet conditions for temperature and water content. These are imposed on the inner wall (intrados) and the outer wall (extrados), as follows:

$$\left\{\begin{array}{lcll} T & = & T_{\text{int}}, & \text{on} \quad \Gamma_{\text{ext}}, \\ T & = & T_{\text{ext}}, & \text{on} \quad \Gamma_{\text{int}}, \end{array}\right. \quad \text{and} \quad \left\{\begin{array}{lcll} C & = & C_{\text{int}}, & \text{on} \quad \Gamma_{\text{ext}}, \\ C & = & C_{\text{ext}}, & \text{on} \quad \Gamma_{\text{int}}. \end{array}\right. \tag{3.15}$$

With regard to mechanical BCs, axisymmetric conditions are specified at the lateral boundaries of the RSV: this implies that normal displacements are assumed to be zero on each lateral face. Furthermore, vertical displacement is assumed to be blocked on the inner face of the RSV, while a uniform vertical displacement is used on the upper face. The set of boundary conditions with a visualization of the mechanical mesh is illustrated in Figure 3.9.

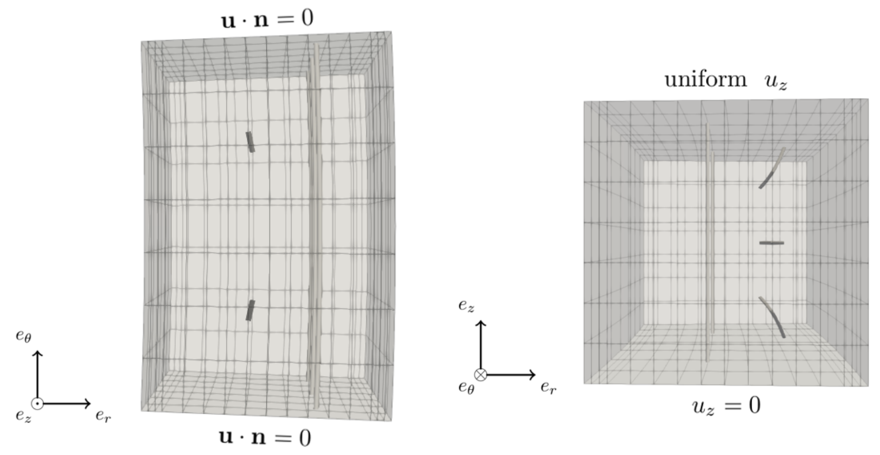


Figure 3.9: BCs for the mechanical problem visualized on the HF mechanical mesh.

3.3.5 A perspective on these model choices in the context of R&D's efforts

The establishment of the mechanical model is based on research efforts spanning multiple stages. This involves several components, including the development of a constitutive equation, the selection of a modeling approach for prestressed concrete involving 3d-1d coupling, and the utilization of a standard enclosure section for mechanical calculations. The choice of a constitutive equation is a critical aspect of developments concerning these materials. Historically, the initial model employed was the `BETON_UMLV` concrete behavior [EDF24]. This model relied on the simulation of two creep models: primary creep, developed from various studies, and drying creep, inspired by Bazant's work. For primary creep, the methodology originally developed to predict long-term deformations under uniaxial stress was adapted. The generalization of this model to account for multiaxial stress states is achieved through the introduction of an arbitrary, constant, or nearly equal Poisson's ratio for creep, compared to the elastic Poisson's ratio ν.

Two issues existed with this constitutive equation: first, the asymptotic behavior overestimated creep, and second, control over the apparent creep coefficient was a challenging issue. Therefore, the basic creep model shifted towards the `BETON_BURGER` constitutive equation [EDF24], on which we already drew our attention. It should be noted that constitutive equations are still an active research area. For instance, Bazant's model [BC85] for drying creep involves an absolute value that can lead to an overestimation of creep during humidity cycles. Furthermore, the choice of studying a representative volume is also a central topic of research discussions. Since the early 2000s, the selection of such an approach has been made and refined. An important trade-off must be made between computational cost, geometric representativeness, and appropriate conditions. Geometric restrictions must be coupled with a suitable choice of applied BCs to ensure a representative simulation. Reevaluations of geometry are often performed based on simulation capabilities or new research findings [Bou16].

Finally, the choice of a numerical simulation with prestressed cables using coupling and analytical tensioning represents a suitable approach resulting from extensive engineering experiments. Historically, fixed nodal efforts were often applied to concrete to model cables. The coupling between cables and concrete allows for a better consideration of prestress loss since the efforts are not necessarily fixed. It becomes much easier to represent the problem nonlinearities. Moreover, the engineers' chosen approach enables the application of analytical tension profiles (provided by official regulations [Règ92]) in the cables as the starting point for simulations. Alternative choices could have been made, such as incorporating numerical calculations in the cables to determine tension profiles. However, this would require numerical calculations to ensure the convergence of nonlinear simulations (contact-friction), especially to account for anchor retreat.

Chapter 4

Numerical investigations on the ROM procedure for structural mechanical problems with internal variables

4.1 Introduction

In this section, we present a validation of the methodology outlined in Chapter 2, applying it to the two validation cases presented in Chapter 3. The objective of this chapter is to Substantiate the different components of the methodology, with the ultimate goal being the industrial case of prestressed concrete with parametric variability (theoretical framework in Section 3.3 and numerical experiments in Section 4.3). The numerical results are presented coherently and progressively towards this end.

First, the focus is on validating the construction of a ROM in the case of single-modeling problem (see Section 4.2). This process is divided into two parts: first, the solution reproduction problem (see Section 4.2.3), aiming to reproduce a HF problem with a reduced problem for the same parameter; second, the parametric problem tests the greedy process on the parametric manifold (see Section 4.2.4). In a preliminary examination, the focus is on verifying various components of the procedure, encompassing creation of both a reduced basis and mesh. Additionally, within the single-modeling framework, we introduced an error indicator that we seek to assess its validity on this test case. The solution reproduction problem provides insights into the ROM's quality by observing various classical metrics. We focus on the projection error, representing the error in projecting HF snapshots onto the reduced space (i.e. the best possible error), and the approximation error, representing the error between the snapshots and the reduced solution obtained after calling the reduced solver. These metrics pertain to the quality of approximation. The second set of metrics aims to examine the effective computational gain achieved when using the reduced solver. This includes the percentage of selected elements, providing an indication of acceleration in terms of computation time. However, the computational time dependency is not linear with mesh size. Solver type and implementation can indeed impact the computational cost depending on mesh size. Besides, in an industrial code, the code may not always be optimized for smaller meshes, for instance, leading to constant computational costs such as memory allocation. A clearer way to estimate CPU computational gains is to work with speedups, defined as the ratio of solver calls:

$$\text{speedup} = \frac{\text{ROM CPU cost}}{\text{HF CPU cost}} \tag{4.1}$$

The parametric tests aim to apply greedy methods to obtain a reduced basis and test the same metrics on the model obtained at the end of the procedure. Therefore, in Section 4.2, we numerically validate the entire procedure on the single-modeling case, by illustrating the method's effectiveness on a simpler instance of a single-modeling mechanical problem with internal variables. Subsequently, in Section 4.3, we seek to extend the validation to the multi-modeling case, introducing information on the reconstruction of quantities of interest (QoI) obtained by post-treatment procedures. The ultimate goal of the thesis project is the parameter tuning, making it imperative that the quantities of interest from the physical problem are accurately reconstructed by the ROM.

All the numerical provided in this Chapter were obtained by relying on FE simulation executed over a commodity Linux workstation (RAM 32 GB, Intel i7-9850H CPU 2.60 GHz x 12).

The theoretical and numerical elements presented in this section are derived from online preprints, specifically from the following works:

- ◇ [AAB$^+$24b] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. International Journal for Numerical Methods in Engineering, 125(4):e7385, 2024.,
- ◇ [AAB$^+$24a] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, Sylvie Michel-Ponnelle, and Tommaso Taddei. Projection-based model order reduction for prestressed concrete with an application to the standard section of a nuclear containment building. *arXiv preprint arXiv:2401.05098*, 2024.

4.2 Single-modeling approach: application to an elastoplastic analysis of a plate with a hole

4.2.1 Definition of the parametric manifold

For the numerical tests, we treat a strain hardening parameter a_{pui} and the Poisson's ratio ν as varying parameters:

$$\mu = \begin{bmatrix} \nu \\ a_{\text{pui}} \end{bmatrix} \in \mathbb{R}^2.$$

The parameter compact is defined as the Cartesian product of parameter intervals $\mathcal{P} = \mathcal{P}_\nu \times \mathcal{P}_{a_{\text{pui}}} = [0.21, 0.3] \times [0.1, 1000]$. At last, we introduce the training set, which is the discrete counterpart of this compact that we shall consider, denoted as $\Theta_{\text{train}} = \Theta_{\text{train},\nu} \times \Theta_{\text{train},a_{\text{pui}}}$. To assess the methodology, we conducted numerical tests in several steps, each dedicated to validating specific features of the designed approach. We consider the scenario of an applied constant load, placing us in a static context. In line with a quasi-static approach commonly adopted in such cases, we opt for proportional loading. In this configuration, time intervals are treated as pseudo-times. Regarding the construction of the reduced basis, we regard this pseudo-time as a physical time scale, and the snapshots taken at this point are also utilized for building the basis. This approach aims to validate the various components of our methodology. However, a potential limitation lies in considering only proportional cases. Therefore, it is relevant to explore temporal variations in loading to verify that it does not compromise the applicability of the method (see Section 4.2.5).

As described in the methodology section, the first step of the validation is the processing of a solution reproduction problem, which illustrates the interest of data compression and the construction of a reduced mesh in terms of CPU cost, while maintaining a quality in the approximation of the solution. Afterwards, we shall discuss two parametric cases: first, we consider a case with a scalar parameter; second, we consider the case of a two-dimensional parameter.

$\lvert\Theta_{\text{train},\nu}\rvert$	$\lvert\Theta_{\text{train},a_{\text{pui}}}\rvert$	K
1	1	20

(a) Solution reproduction problem.

$\lvert\Theta_{\text{train},\nu}\rvert$	$\lvert\Theta_{\text{train},a_{\text{pui}}}\rvert$	K
20	1	10

(b) Parametric Problem for $\mu = \nu$.

$\lvert\Theta_{\text{train},\nu}\rvert$	$\lvert\Theta_{\text{train},a_{\text{pui}}}\rvert$	K
20	20	10

(c) Parametric Problem for $\mu = (\nu, a_{\text{pui}})$.

Figure 4.1: Summary of the size of the training sets and the number of timesteps (K) used for the different test cases.

The choice of the parameter subset size ($|\Theta_{\text{train}}|$) and the number of time steps (K) are indicated in Figure 4.1. We briefly outline here the motivation for these different decisions. As far as the temporal discretization is concerned, the calculation converges after ten time steps ($K = 10$) for all test cases considered. First, when assessing the ROM methodology on the solution reproduction problem (see Section 4.2.3), we opt to rely on a grid twice as fine as in the parametric case. Such a choice provides a better visualization of the quality of the ROM, notably in order to test the correlation between the devised error indicator and the approximation error. As explained previously, we opt to start from a 2d Cartesian grid of parameters for all the numerical experiments. Our case is such that the greedy algorithm converges in less than $n_{\text{it}} = 10$ iterations (see detailed numerical results in Section 4.2.4). Therefore, we choose to consider about twenty parameters in each direction of the grid, which means $|\Theta_{\text{train}}| = 40$.

4.2.2 Error metric

To assess the quality of the ROM build, we must be able to measure the latter's quality. One initial idea is to look at projection errors. This refers to the difference between the HF snapshots projected onto the reduced space and the HF snapshots. It assesses the best accuracy with which the reduced model represents important features of the system. We are interested in an error in time, and formulated on a form of relative error, i.e. by comparison with a reference measurement, here the norm of the HF model output. In this specific test case, the norm used is the energy norm at the centroid ($\|\cdot\|_{a_{\overline{\mu}}}$) presented before (cf. Equation (2.44)), and we consider a uniform discretization for the time steps, which leads to the following reformulation of the expression for $\mu \in \mathcal{P}$:

$$
\begin{aligned}
E_{u,\mu}^{\text{proj,avg}} = \frac{\sqrt{\sum\limits_{k=1}^{K} \Delta t^{(k)} \left\| u_{\mu}^{\text{hf},(k)} - \Pi_{\mathcal{Z}_u} u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \Delta t^{(k)} \left\| u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}} &= \frac{\sqrt{\sum\limits_{k=1}^{K} \Delta t^{(k)} \left\| \Pi_{\mathcal{Z}_u^{\perp}} u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \Delta t^{(k)} \left\| u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}} \\
&= \frac{\sqrt{\sum\limits_{k=1}^{K} \frac{1}{K} \left\| \Pi_{\mathcal{Z}_u^{\perp}} u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \frac{1}{K} \left\| u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}} \\
&= \frac{\sqrt{\sum\limits_{k=1}^{K} \left\| \Pi_{\mathcal{Z}_u^{\perp}} u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \left\| u_{\mu}^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}
\end{aligned}
$$

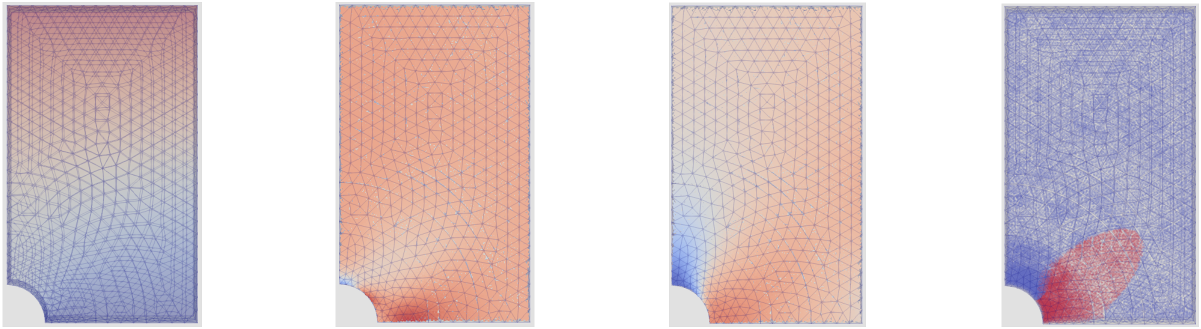
The approximation error measures the difference between the output of the full order model and that of the ROM. A low approximation error indicates a good ability of the ROM to capture the behavior of the complete system during online calculation. We introduce the time-averaged projections errors ($E_{u,\mu}^{\text{proj,avg}}$) and approximation errors ($E_{u,\mu}^{\text{app,avg}}$) on the displacement for any $\mu \in \mathcal{P}$:

$$
E_{u,\mu}^{\text{proj,avg}} = \frac{\sqrt{\sum\limits_{k=1}^{K} \left\| \Pi_{\mathcal{Z}_u^{\perp}} u^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \left\| u^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}} \quad \text{and} \quad E_{u,\mu}^{\text{app,avg}} = \frac{\sqrt{\sum\limits_{k=1}^{K} \left\| u^{\text{hf},(k)} - \widehat{u}^{(k)} \right\|_{a_{\overline{\mu}}}^{2}}}{\sqrt{\sum\limits_{k=1}^{K} \left\| u^{\text{hf},(k)} \right\|_{a_{\overline{\mu}}}^{2}}} \tag{4.2}
$$

4.2.3 Solution Reproduction Problem

First, we present numerical results for a fixed configuration of parameters, namely for the centroid $\overline{\mu} \in \mathcal{P}$ to validate our ROM strategy. In Figure 4.2, we display the snapshots at the last time step for the problem we wish to reproduce. The vertical displacement component (u_y), two stress tensor components (σ_{xx} and σ_{yy}) and the plastic zone are depicted. In this scenario, we examine a situation where the nonlinearity is not prominently activated. The plastic zone, indicating nonlinear behavior, remains confined in proximity to the hole, while the majority of the material remains in an elastic regime.

Figure 4.3 represents the POD eigenvalues obtained for the displacements and the stresses. We notice that the decays of the eigenvalues have a similar profile, although the decay of the eigenvalues is slightly faster for the displacement field than for the stress field. The plot of the projection errors as a function of the number of modes used to build the reduced space highlights this capacity to better estimate the displacement trajectory for a smaller number of modes. This



(a) Component of the displacement u_y.

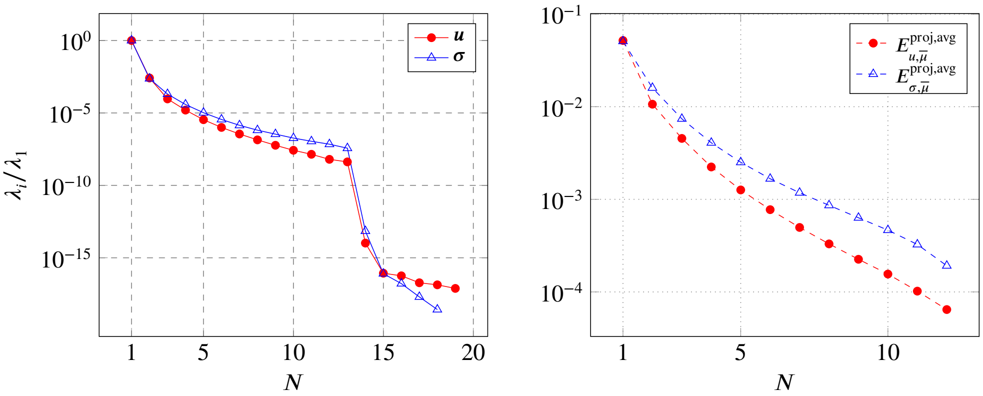
(b) Component of the stress σ_{xx}.

(c) Component of the stress σ_{yy}.

(d) Plastic region (red) vs elastic region (blue).

Figure 4.2: Visualization of different snapshots of the FE solution for HF calculation for the final time step. The physical parameters used correspond to those used in the study for the non-parametric case. We provide here a displacement component (vertical displacement), two components of the Cauchy stress tensor and the description of the plastic (red) and elastic (blue) zones for this numerical example.

suggests that in order to get projection errors in displacements and stresses at a given order of magnitude, it is mandatory to have more stress modes than displacement modes. A key aspect of our reduction methodology lies in the separate construction of two reduced bases, one for displacements and the other for stresses. While this independent approach simplifies our procedure, it also raises the complexity of predicting *a priori* the projection of a displacement solution into the stress space. An undesirable situation we wish to avoid is, for instance, that of obtaining a displacement response whose constitutive equation integration yields a regime that our reduced stress space is unable to accommodate. To overcome this limitation, our approach relies mainly on the idea of considering larger stress spaces. However, we are aware that alternative perspectives could be formulated in order to have a more systematic and robust numerical method to provide an adequate stress space for a given reduced displacement space.



(a) POD eigenvalues.

(b) Projection errors.

Figure 4.3: Solution reproduction problem: (a) behavior of the POD eigenvalues for displacement ($\mathbf{u}$) and stress ($\boldsymbol{\sigma}$) for several values of $N(N = N_u$ for $\mathbf{u}$ and $N = N_\sigma$ for $\boldsymbol{\sigma}$; b) behavior of the average projection errors) .

In line with this remark, we start the presentation of numerical results on the ROM with the error indicator. As a quick reminder, the error indicator's function is to gauge the accuracy of the ROM approximation, emphasizing the importance of maintaining a strong correlation between the indicator and the actual quality of the approximation. Figure (4.4) displays a good correlation between the the error indicator used and the approximation error on the solution fields. We point out that, in every case reported here, we have chosen to deal with all the available stress modes. Indeed, for extremly underresolved reduced spaces, the error indicator is found to be inaccurate. Since the construction of our error indicator relies on an approximation of the dual norm using the decomposition of the stress field on the space of stress modes, the correlation between the error indicator and the approximation errors may be slightly degraded for too coarse approximation spaces. This choice of treating all the stress modes does not raise overfitting problems during the Gappy-POD since we have a number of modes lower than the number of elements selected during the hyper-reduction procedure, in our quite simple case. From a practical standpoint, this choice allows us not to have to play with the ratio between the two compression tolerances ($\varepsilon_{\mathrm{POD},u}$ and $\varepsilon_{\mathrm{POD},\sigma}$) for the construction of the reduced problem.

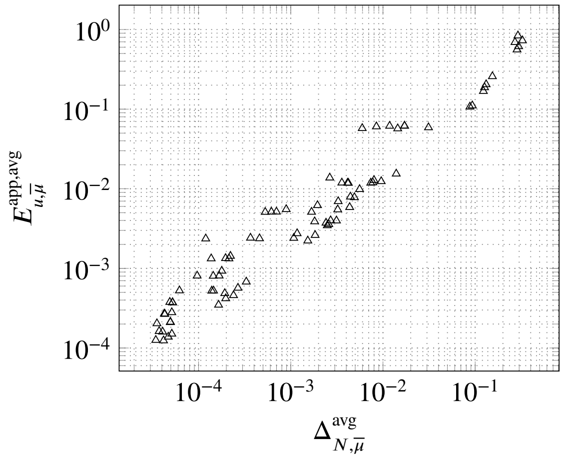


Figure 4.4: Correlation between the approximation error on the displacement ($E_{u,\overline{\mu}}^{\mathrm{app,avg}}$) and the error indicator ($\Delta_{N,\overline{\mu}}^{\mathrm{avg}}$) for the solution reproduction problem.

We have built reduced models for various numbers of modes (compression of the solution space) and various hyper-reduction parameters (size of the reduced mesh). The aim of investigating this grid of hyperparameters of the reduced model is multifaceted. First, it enables to investigate a wide range of approximation errors. Indeed, the quality of the approximated solution depends on the approximation quality of the integrals involved in the problem (δ) but also on the approximation quality of the trajectory (N_u). This variation allows us to highlight the correlation between the approximation error on the displacement field and the error indicator that we have presented (cf. Figure 4.4, and Colormaps 4.7a and 4.7b). Moreover, for a fixed number of modes, the projection error constitutes a theoretical lower bound that we wish to be able to reach by solving the reduced problem. However, the hyper-reduction process introduces a new approximation. In Figure 4.5, we illustrate that the approximation error tends towards the projection error for small δ values, while a less restrictive parameter degrades the solution ($\delta = 10^{-1}$ for example). The slight differences between approximation and projection errors between the last two δ values comes from the fact that we hit the tolerance of the iterative Newton algorithm used in the HF solver (which is chosen as the same as in the reduced solver).

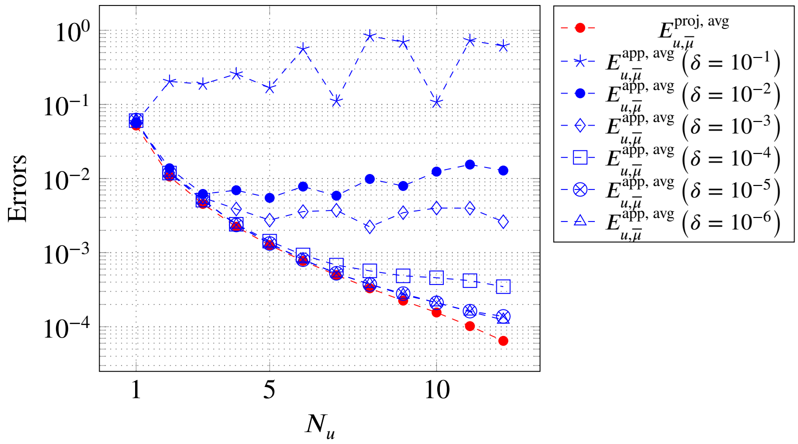


Figure 4.5: Comparison between the approximation and the projection errors with respect to the number of displacement modes N_u for the solution reproduction problem. Approximation errors provided here have been computed for different values of the hyper-reduction parameter δ (EQ tolerance).

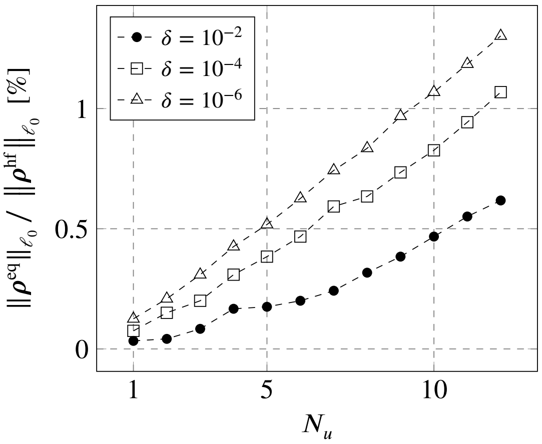
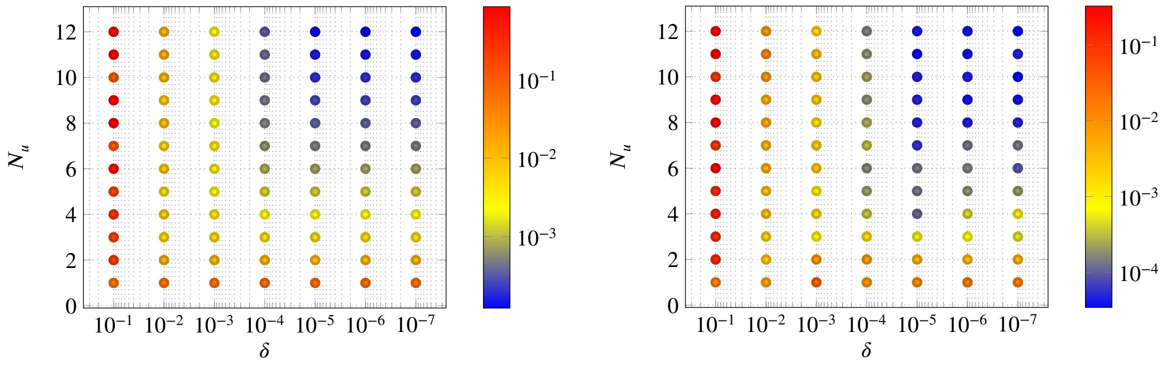


Figure 4.6: Percentage of three-dimensional selected elements depending on the size of the primal basis (N_u) and the EQ tolerance δ for the solution reproduction problem.

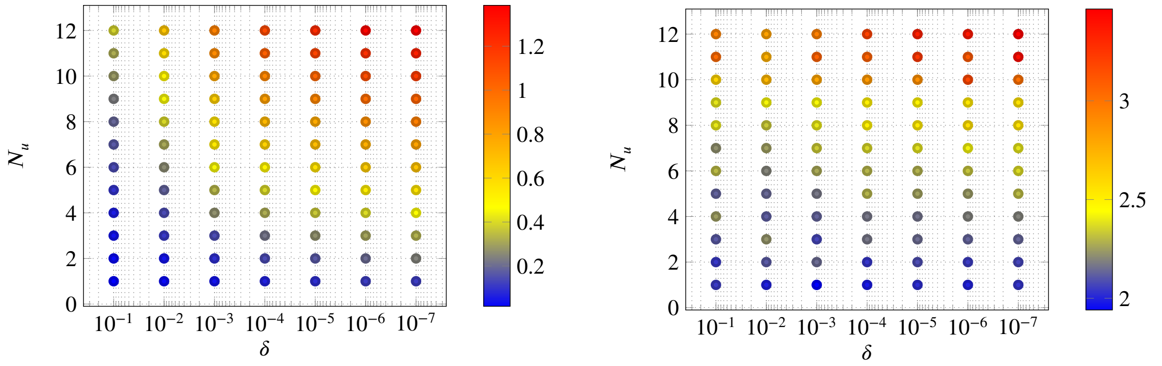
Much more, we observe that the empirical quadrature procedure is able to significantly reduce the size of the mesh used for online calculations. We keep at most a few percent of the number of elements in the HF mesh. We thus drastically reduce the computational cost compared to a HF problem. Indeed, the cost of a reduced problem represents only a few percent of the cost of the HF calculation. The computational cost reduction, correlated to the number of selected elements (cf. Colormap 4.7c), depends both on the number of selected modes and on the hyper-reduction parameter that we choose.

It should be noticed in the following case that the ratio between the computational cost of the reduced problem and the percentage of selected elements are not strictly correlated, even if the two quantities follow the same tendency. Indeed, from an algorithmic point of view, the reduction of the mesh is not the only operation involved between a HF computation and a reduced computation, since the projection on the modes and the hyper-reduction entail a modification of the size of the system, but also of the conditioning of the latter (this can lead to more Newton iterations for a reduced computation for example). Furthermore, the implementation has been done in an industrial code where fixed costs related to verification and memory allocation processes are necessary whatever the computation. Nevertheless, in Figure 4.7, we provide a numerical validation that the percentage of selected elements gives us a good hint on the gain in terms of computational cost.

We have observed that the approximation error on the stresses follows the same pattern as the approximation error on the displacements on the hyperparameter grid ($N_u \times \delta$). This comment brings us to report only approximation errors on displacements in this contribution.



(a) $E^{\text{app,avg}}_{u,\overline{\mu}}$ (b) $\Delta^{\text{avg}}_{N,\mu}$



(c) $\|\boldsymbol{\rho}^{\text{eq}}\|_{\ell_0} / \ \|\boldsymbol{\rho}^{\text{hf}}\|_{\ell_0}$ [%] (d) ROM CPU cost / HF CPU cost [%]

Figure 4.7: Solution reproduction problem: colormaps of: (a) approximation errors, (b) error indicators, (c) percentage of selected elements, (d) percentage of CPU time for different size of reduced order basis and hyper-reduction parameters.

4.2.4 Parametric problem

Parametric case for $\mu = \nu$

In this section, we consider the variation of a single parameter (ν), for a training set of $|\Theta_{\text{train},\nu}| = 20$ values of this parameter. The numerical results presented here and in the last sub-section were performed for a smaller number of time steps for the sake of efficiency (see parameters in Table 4.1).

We were able to test the greedy approach on this single parameter nonlinear case where we fixed a given number of iterations ($n_{\text{it}} = 5$). For the studied example, the algorithm has reached its convergence for the following number of iteration. All the examples reported here have been carried out for the tolerance $\varepsilon_{\text{POD},u} = 10^{-5}$, which ensures a good approximation error on the explored parameters. The evolution of the maximum of the error indicator (cf. Figure 4.8) for several hyper-reduction parameters shows a convergence after a few iterations. The plateau reached by the error indicator differs with the accuracy of the approximation of the integrals of the problem.

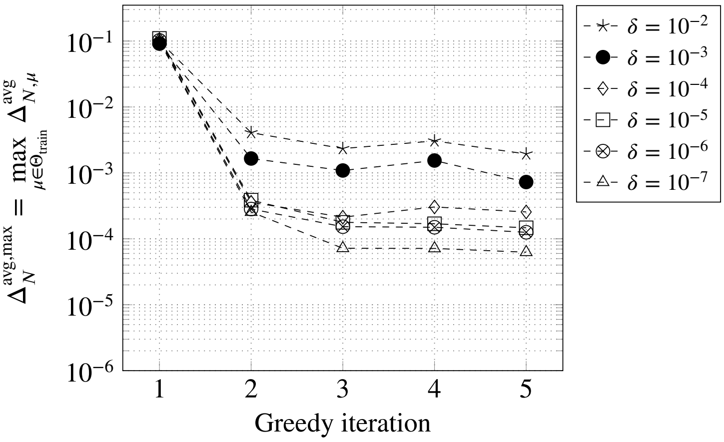


Figure 4.8: Maximum of the time-averaged error indicator over the training set depending on the Greedy iteration for the parametric problem $\mu = \nu$.

We report the computational costs associated with the reduced solver by giving the speedup, where the HF cost is the computational time of solving the HF problem whereas rom cost is the online cost of evaluating the problem. In Figure 4.9, we notice that the gain in computation time decreases with each iteration as the percentage of selected elements increases with the number of HF problems to be estimated and the number of modes to be included in the reduced basis. Nevertheless, for this single-parameter problem, the speedups obtained are always higher than 10 or even 15, which implies a drastic decrease of the computation time for the model evaluation. Moreover, the parametric manifold is in our case very well approximated after a small number of iterations. The plateau observed in Figure 4.8 is reached after a few iterations and shows that for the given tolerance of hyper-reduction and the desired precision in the compression of the base, there is no more gain in exploring a new parameter value.

We notice on the reduced meshes obtained at the end of the numerical procedure (cf. Figure 4.10) that the selected elements are mainly located around the hole, which matches the region where the material enters a nonlinear regime (plastic regime). As we would expect, few elements are selected in the areas where the behavior is purely elastic (linear). Indeed, one can expect that the algorithm will select more elements in areas where there is more energy while rewriting the elementary contributions. If there is a plastic zone and an elastic zone, the selection will therefore focus on the plastic regions. If the whole geometry is in the plastic region, elements can be selected from the entire mesh, with the emphasis on the areas that plasticized first. However, it is worth mentioning that the percentage of elements selected will naturally increase if the plastic zone is larger. The methodology is adapted for a case where the behavior over the whole geometry is nonlinear. In this case, greater attention must be paid to the choice of the δ hyperparameter. In Figure 4.11, we provide a verification of the interpretation formulated above. We denote σ_y^{ref} the

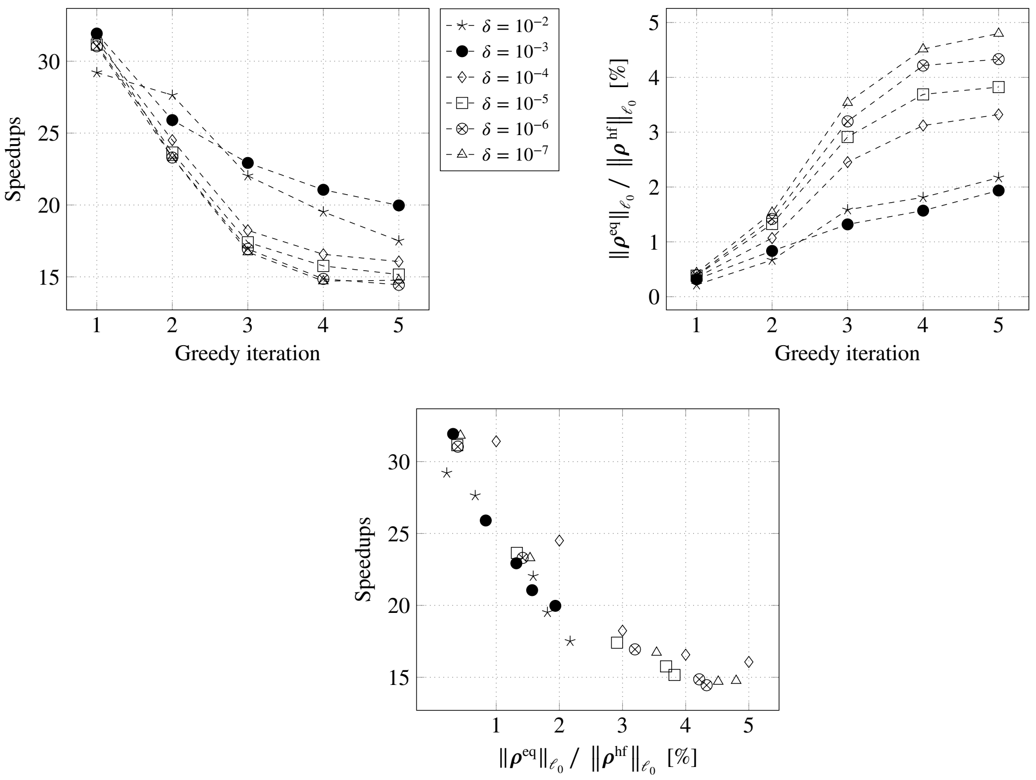
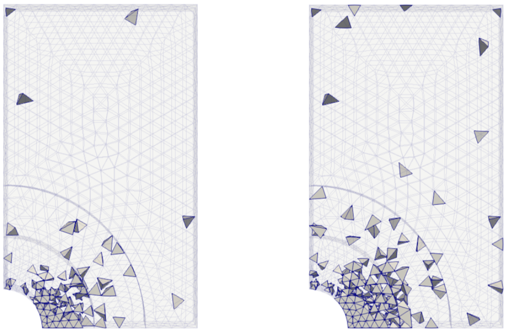


Figure 4.9: Parametric problem $\mu = \nu$: informations (speedups and percentage of three-dimensional selected elements $\|\boldsymbol{\rho}^{\text{eq}}\|_{\ell_0} / \left\|\boldsymbol{\rho}^{\text{hf}}\right\|_{\ell_0}$ [%]) through the Greedy iterations ($\varepsilon = 10^{-5}$) for different values of δ.

elastic limit used in the nominal study case. By varying this parameter, we can create a regime with a large plastic region (almost the entire geometry) and a regime with a small plastic region. The visualization of reduced meshes in relation to the plastic region in these cases confirms our hypotheses. We therefore believe that the procedure is still effective in the case of a fully plasticized domain.



(a) Reduced mesh for $\delta = 10^{-2}$.

(b) Reduced mesh for $\delta = 10^{-7}$.

δ	10^{-2}	10^{-7}
Speedup	17.51	14.77
$\|\boldsymbol{\rho}^{\text{eq}}\|_{\ell_0} / \left\|\boldsymbol{\rho}^{\text{hf}}\right\|_{\ell_0}$ [%]	2.17	4.80
N_u	23	22

(c) Summary of outputs of the hyper-reduction procedure.

Figure 4.10: Parametric problem $\mu = \nu$: hyper-reduced ROM and selected elements at the end of the POD-Greedy procedure ($\varepsilon_{\text{POD},u} = 10^{-5}$) for two different values of δ.

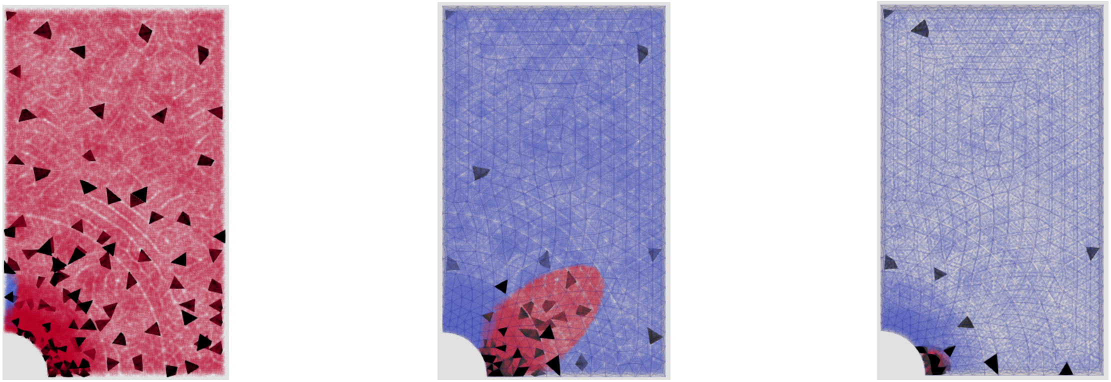

(a) $\sigma_y = \frac{2}{3}\sigma_y^{\text{ref}}$ (b) $\sigma_y = \sigma_y^{\text{ref}}$ (c) $\sigma_y = \frac{4}{3}\sigma_y^{\text{ref}}$

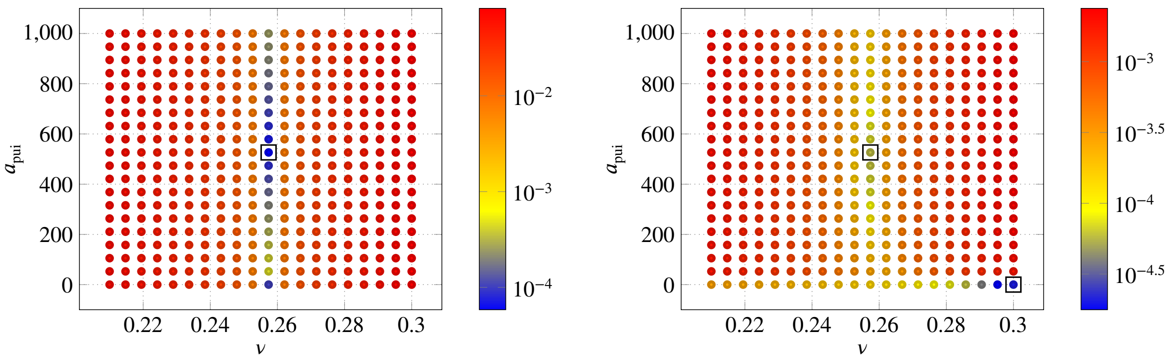
Figure 4.11: Visualization of reduced meshes and the plastic region (obtained for the HF calculation) for a variation in the elastic limit. The red zone indicates the area where the material is in the plastic (nonlinear) regime, while the blue zone corresponds to the purely elastic zone. The meshes selected for the reduced mesh appear in black. The hyper-reduction provided here has been carried out for a solution reproduction problem where 20 time steps are used in the HF calculation and for reduction parameters $(N_u, \delta) = (4, 10^{-6})$.

Parametric case for $\mu = (\nu, a_{\text{pui}})$

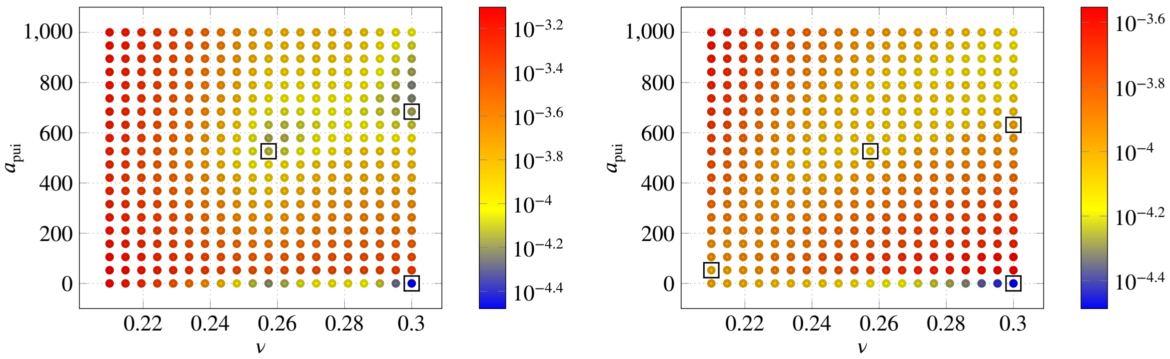
Finally, we provide a numerical example for two parameters. We address a training set of size $|\Theta_{\text{train}}| = 400$. We report here the results for tolerances $\varepsilon_{\text{POD},u} = 10^{-5}$ and $\delta = 10^{-7}$. The choice of the hyper-reduction parameter is chosen here of the same order of magnitude as the Newton tolerance for HF computation. In Figure 4.12, we present the evolution of the error indicator we compute over the greedy iterations. By comparing, for instance, the colormap at the second iteration and at the third iteration, we notice that the sampling of a parameter leads to a decrease of the indicator value in the neighborhood of the given parameter. Moreover, we were interested in the correlation of the error indicator with the error indicator especially with out-of-sample parameters. To this end, we defined a sub-grid of 25 points, (5×5 Cartesian grid of the parameters), on which the HF calculations were performed in order to dispose of the projection error. In Figure 4.13, we show the profiles obtained for the error indicators and for the approximation errors on the parameters chosen for the test. It appears that the error indicator seems to follow the behaviour of the approximation error.

We point out that we only provide results on the first iterations because we have limited ourselves to a small number of iterations as the problem is sufficiently well approximated in a short time. It would therefore not be relevant to compare relative errors where the variation between the parameters becomes insignificant.

We report the offline costs associated to the construction of the ROM in Table 4.1. We focus on the computational costs involved in constructing the EQ rule for three-dimensional terms, and in constructing the reduced displacement basis. Indeed, the number of two-dimensional elements is lower than that of three-dimensional elements, and therefore absorbs the offline computation cost. For the basis computation, we use the same number of HF snapshots (in displacements and stresses). As a result, the Gram matrix constructed for the SVD process has the same size for both fields (even if the computational cost of constructing the matrix varies). We notice that



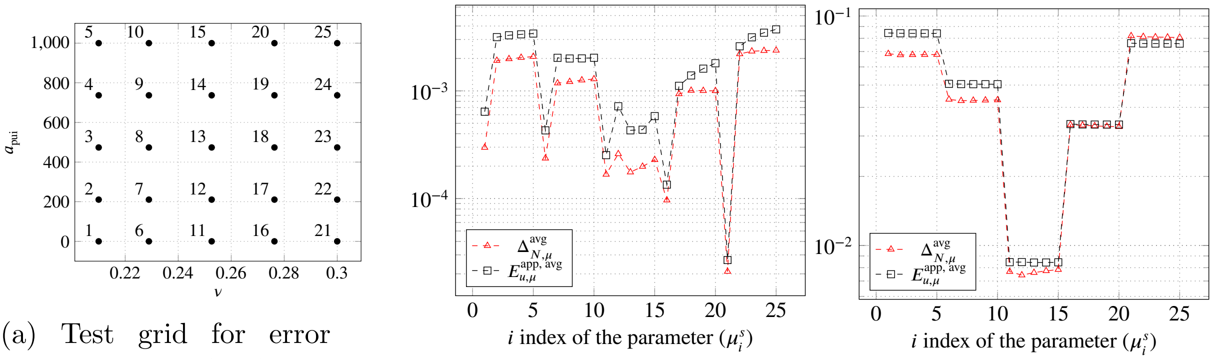
(a) Greedy iteration n°1 (b) Greedy iteration n°2



(c) Greedy iteration n°3 (d) Greedy iteration n°4

Figure 4.12: Parametric problem $\mu = (\nu, a_{\text{pui}})$: colormaps of the time-averaged error indicators and selected parameters (points squared in black) for every Greedy iterations.

the computational cost of constructing the reduced basis remains reasonable in the context of our study, i.e. lower than the computational cost of a HF calculation. In this application case, the offline computational cost of generating the empirical quadrature quickly becomes greater than the cost of using an HF computation. This observation can be explained by several factors. First, the construction and solution of the optimization problem is not performed incrementally (unlike our approach for the reduced basis). An incremental optimization approach based on the knowledge of an empirical quadrature at the previous iteration is one way of preventing the offline computation cost from exploding at each iteration, and it seems a suitable way of controlling the CPU computation cost. Second, the optimization problem only depends on the number of elements in the mesh and the number of snapshots used. In our case, the computational cost of an HF problem is quite low, and is therefore quickly caught up by the cost of solving the optimization problem. However, for problems with more pronounced nonlinearities, or larger mesh sizes, it is fairly likely that the ratio between this offline computation cost and the HF computation cost will shift, as the latter will be much higher. This interpretation is consistent with pre-existing results in the literature on problems where HF computation costs are of a different order of magnitude. Finally, although the hyper-reduction method provides convincing results in the online phase, more work needs to be done to gain better control over the computational cost of the optimization process.



(a) Test grid for error computation and indexing of the test set.

(b) Greedy iteration n°2

(c) Greedy iteration n°1

Figure 4.13: Parametric problem $\mu = (\nu, a_{\text{pui}})$: correlation between the error indicator and the approximation errors depending on the numerotation of the test parameters (repartition shown in 4.13a).

Iteration	**CPU cost ($\boldsymbol{\rho}^{\text{eq},\Omega}$) [s]**	**CPU cost ($\mathbf{Z}_u$) [s]**	**CPU cost (HF solver) [s]**
1	1.11	$6.47 \cdot 10^{-2}$	$9.72 \cdot 10^{1}$
2	$1.73 \cdot 10^{1}$	7.89	$7.25 \cdot 10^{1}$
3	$1.12 \cdot 10^{2}$	$1.93 \cdot 10^{1}$	$9.81 \cdot 10^{1}$
4	$2.55 \cdot 10^{2}$	$2.82 \cdot 10^{1}$	$9.31 \cdot 10^{1}$
5	$6.07 \cdot 10^{2}$	$3.69 \cdot 10^{1}$	$9.22 \cdot 10^{1}$
6	$1.16 \cdot 10^{3}$	$5.10 \cdot 10^{1}$	$9.69 \cdot 10^{1}$

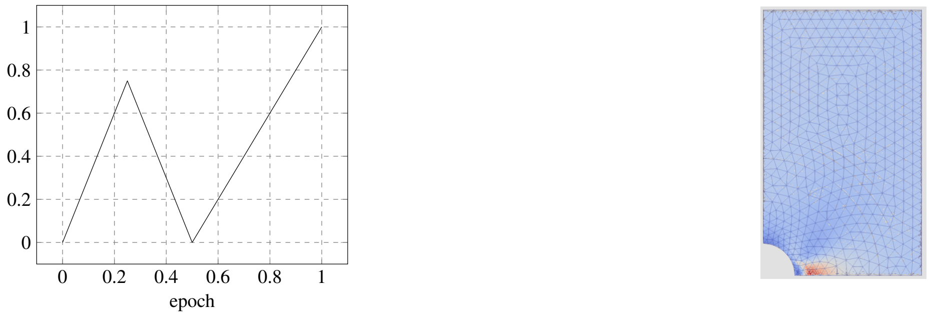
Table 4.1: CPU Offline costs for the greedy procedure in the case of the multi-parametric problem $\mu = (\nu, a_{\text{pui}})$.

4.2.5 Extension to a loading-unloading case

We present here the case of a loading-unloading numerical experiment for an elastoplastic case. Indeed, in the previous sections, we illustrated the interest of our reduction procedure for elastoplasticity problems for quasi-static cases with linear loading. To numerically assess whether the adaptive procedure remains valid in slightly more complex cases, we decided to investigate a load-unload case leading to an expansion of the yielding surface. Since loading is more complex, the case presented here is solved in 40 time steps.

Provided that the error indicator constructed remains correlated with the approximation errors in displacement, the numerical procedure presented above remains valid. We therefore examine a case, using the previous parameters, where the (previously linear) loading history includes a loading and then a unloading phase. These steps are detailed in the Figure 4.14, where we also show a non-zero stress state in the middle of the loading process, illustrating that we are in a residual stress setting. We only focus on the non-parametric case to highlight the correlation between the error indicator and the approximation error (so as not to overload the document).

To shed light on the correlation between approximation error and error indicator, we choose to monitor these quantities over the course of the loading history. In the Figure 4.15, we display the approximation errors in displacement at each time step and the error indicators at each time



(a) Proportionality coefficient used to multiply the final load applied as a function of simulation time.

(b) Example of a stress field component (σ_{xx}) in the middle of loading procedure (end of unloading). Values in light blue correspond to zero values, unlike values in dark blue or red.

Figure 4.14: Details of the loading-unloading cases studied: presentation of the proprotionality coefficient by which the final load is multiplied at each time step, and visualization of a stress field component with non-zero terms at the end of unloading.

step. We notice that the magnitudes are correlated over time for several hyper-reduction parameters (and therefore several approximation qualities). We emphasize that it is crucial to ensure that the approximation error in stress is of similar quality to that in displacement, and that the Gappy-POD does not degrade the quality of the reduced solution. If this condition is met (as in the case presented here), the desired correlation is achieved.

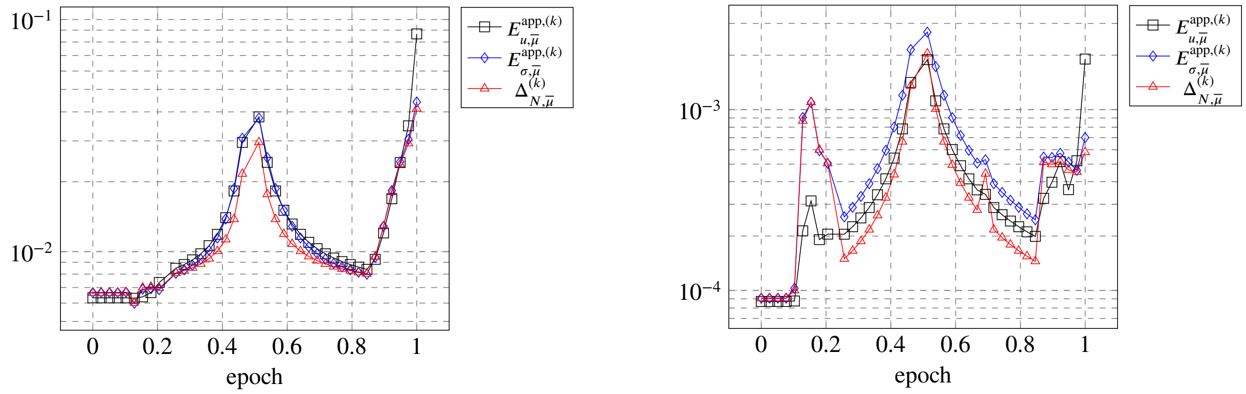
We generated a dataset of reduced models by varying the number of displacement modes and hyper-reduction tolerance. Each pair of parameters then defines a given ROM. In Figure 4.16, we show the set of error indicators and approximation errors at each time step for all parameter pairs (i.e. for several built ROMs). We do indeed maintain a correlation between error indicators and errors.

4.3 Thermo-Hydro-Mechanical (THM) modeling of large concrete structures

4.3.1 Adaptation of the ROM methodology

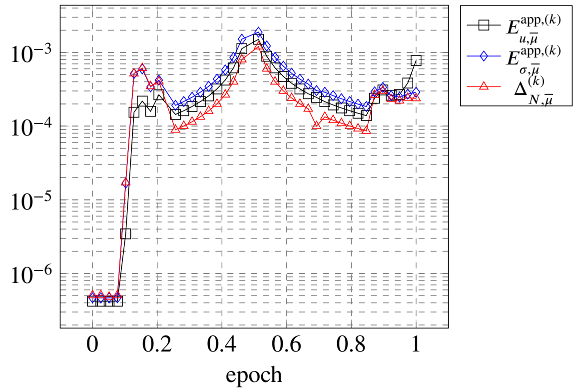
We have developed a hyper-reduced ROM approach to account for nonlinearities in three-dimensional domains. Transitioning to multi-dimensional levels requires some adjustments, primarily at two levels: first, in the construction of the reduced mesh; and second, concerning the basis used for reconstructing the stress or normal forces. It is important to note that these two points are interconnected and rely on the construction of a reduced integration domain.

In the context of our prestressed concrete problem, we have two meshes—one three-dimensional with nonlinear behavior and the other one-dimensional with linear behavior. We choose to hyper-reduce only the mesh associated with concrete, thus retaining the entirety of the bars. No changes are necessary to preserve the kinematic links between them, as the displacement modes are defined on global nodal unknowns, encompassing both 3d and 1d DOFs. Consequently, the displacement modes adhere to the desired kinematic conditions, ensuring the coupling between concrete and



(a) $\delta = 10^{-2}$

(b) $\delta = 10^{-4}$



(c) $\delta = 10^{-6}$

Figure 4.15: Correlation between approximation errors in displacement (relative error at a given time step - $E_{u,\overline{\mu}}^{\mathrm{app},(k)}$) and stress ($E_{\sigma,\overline{\mu}}^{\mathrm{app},(k)}$) with error indicators (error indicator at a given time step - $\Delta_{N,\overline{\mu}}^{(k)}$) over the loading history (unit of time denoted by epoch) for $N_u = 12$ displacement modes and several hyper-reduction tolerances (δ). In the plot, time steps where loading is zero have been removed to avoid normalization problems.

steel. It is important to emphasize the significance of this choice since the **B**-matrix is independent of displacement fields in our framework, $\forall v \in \mathcal{X}^{\mathrm{hf}}$:

$$\begin{aligned}
\mathcal{R}_{\mu}^{\mathrm{hf}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \boldsymbol{\mathfrak{S}}_{\mu}^{(k-1)},\ \mathbf{v}\right) &= \mathcal{R}_{\mu}^{\mathrm{hf,3d}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \boldsymbol{\sigma}_{\mu}^{(k-1)},\ \mathbf{v}\right) + \mathcal{R}_{\mu}^{\mathrm{hf,1d}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \mathbf{N}_{\mu}^{(k-1)},\ \mathbf{v}\right)\\
&\approx \underbrace{\mathcal{R}_{\mu}^{\mathrm{eq,3d}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \boldsymbol{\sigma}_{\mu}^{(k-1)},\ \mathbf{v}\right)}_{\text{hyper-reduced}} + \underbrace{\mathcal{R}_{\mu}^{\mathrm{hf,1d}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \mathbf{N}_{\mu}^{(k-1)},\ \mathbf{v}\right)}_{\text{not hyper-reduced}}\\
&:= \mathcal{R}_{\mu}^{\mathrm{eq}}\left(\mathbf{u}_{\mu}^{(k)},\ \mathbf{u}_{\mu}^{(k-1)},\ \boldsymbol{\mathfrak{S}}_{\mu}^{(k-1)},\ \mathbf{v}\right)
\end{aligned}$$

Furthermore, we opt to construct generalized force $\boldsymbol{\mathfrak{S}} = [\boldsymbol{\sigma}, \mathbf{N}]^{\top}$ modes, considering both stresses and normal forces. These quantities are non-homogeneous, necessitating the construction of a suitable scalar product to handle consistent quantities:

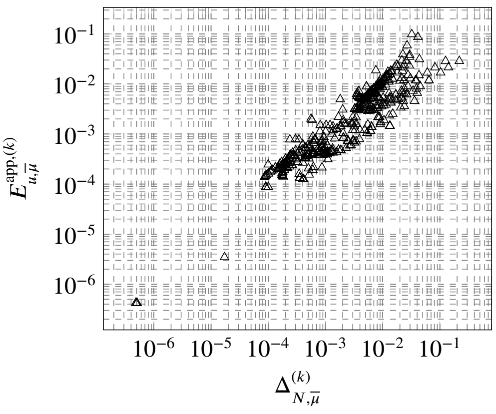


Figure 4.16: Correlation between approximation errors in displacement with error indicators for every time steps, with reduced solutions generated for $N_u \in \{3, 6, 9, 12\}$ and $\delta \in \{10^{-2}, 10^{-4}, 10^{-6}\}$.

$$(\mathfrak{S}_1, \mathfrak{S}_2) = \left(\begin{bmatrix} \boldsymbol{\sigma}_1 \\ \mathbf{N}_1 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\sigma}_2 \\ \mathbf{N}_2 \end{bmatrix} \right)_{[\sigma,N]} = \frac{1}{\lambda_1^{\sigma}} (\boldsymbol{\sigma}_1,\ \boldsymbol{\sigma}_2)_2 + \frac{1}{\lambda_1^{\mathrm{N}}} (\mathbf{N}_1,\ \mathbf{N}_2)_2 ,$$

where λ_1^{σ} (resp. λ_1^{N}) is the largest eigenvalue in the sense of the scalar product ℓ_2 for the stress vectors (normal forces). In summary, in addition to the EQ rule $\boldsymbol{\rho}^{\mathrm{eq}}$ (and the associated reduced mesh), the ROM is made up of two reduced bases, defined thanks to the POD operator detailed in Equation (2.26) as follows:

$$\mathbf{Z}_u = \mathrm{POD}\left\{ \{\mathbf{u}^{\mathrm{hf},(k)}_{\mu}\}_{k=1}^{K},\ (\cdot,\cdot)_2\,,\ \varepsilon_{\mathrm{POD},u} \right\}, \quad \text{and} \quad \mathbf{Z}_{\mathfrak{S}} = \mathrm{POD}\left\{ \{\mathfrak{S}^{\mathrm{hf},(k)}_{\mu}\}_{k=1}^{K},\ (\cdot,\cdot)_{[\sigma,N]}\,,\ \varepsilon_{\mathrm{POD},\mathfrak{S}} \right\}.$$

where $\varepsilon_{\mathrm{POD},u}$ and $\varepsilon_{\mathrm{POD},u}$ are the two corresponding tolerances. Both for the displacements and for the generalized forces, we opted for a scalar product ℓ_2 on the discrete snapshots. One of the reasons for this is the difficulty of finding a suitable norm that is both consistent with the THM problem and easy to implement in `code_aster` (reconsider the need to use an energy norm for the single-modeling case, given by Equation (2.44)). For these applications, we choose to test and validate this compression at the discrete level, which is common practice when applying it to real-world applications. We recall and emphasize that the error indicator developed is limited to the three-dimensional case. As a result, the greedy procedures described subsequently rely on the approximation error and thus, we focus on a strong-greedy approach for building ROMs.

4.3.2 Solution Reproduction Problem

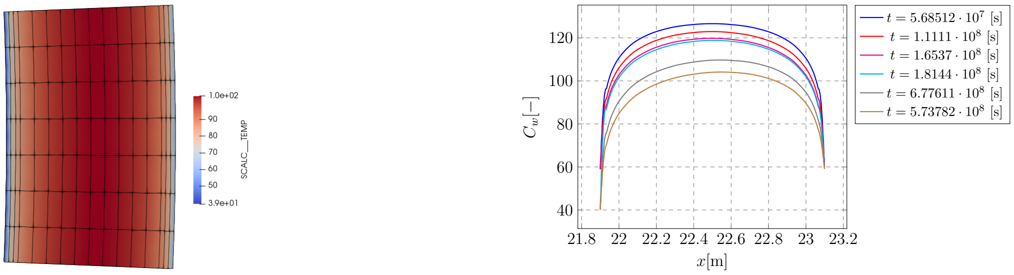
We first perform a validation of the methodology on a non-parametric case. We aim to mimic the HF simulation with our ROM for the same set of parameters. To assess the quality of the reduced model, we introduce several metrics. First of all, since our ROM is founded on a projection onto displacement modes, we introduce displacement approximation errors, at a given time step ($E^{\mathrm{app},(k)}_{u,\mu}$), and averaged over time ($E^{\mathrm{app,avg}}_{u,\mu}$):

$$E_{u,\mu}^{\mathrm{app},(k)} = \frac{\left\| \mathbf{u}_{\mu}^{\mathrm{hf},(k)} - \widehat{\mathbf{u}}_{\mu}^{(k)} \right\|_2^2}{\left\| \mathbf{u}_{\mu}^{\mathrm{hf},(k)} \right\|_2^2}, \quad \text{and} \quad E_{u,\mu}^{\mathrm{app,avg}} = \frac{\sqrt{\sum_{k=1}^{K} \frac{t^{(k)}-t^{(k-1)}}{t_{\mathrm{f}}} \left\| \mathbf{u}_{\mu}^{\mathrm{hf},(k)} - \widehat{\mathbf{u}}_{\mu}^{(k)} \right\|_2^2}}{\sqrt{\sum_{k=1}^{K} \frac{t^{(k)}-t^{(k-1)}}{t_{\mathrm{f}}} \left\| \mathbf{u}_{\mu}^{\mathrm{hf},(k)} \right\|_2^2}}, \tag{4.3}$$

where t_{f} is the final physical time used in the simulation and where $\mathbf{u}_{\mu}^{\mathrm{hf},(k)}$ and $\widehat{\mathbf{u}}_{\mu}^{(k)}$ are respectively the solution at the k-th timestep obtained when using the HF model or the ROM for the parameter μ. For the simulations reported below, we simulate a physical time of around 18 years.

HF problem

In this section, we present the HF problem we wish to reproduce. As previously stated, we are only seeking to reduce the mechanical calculation in our THM coupling. To this end, we rely on a thermo-hydraulic calculation, which can be viewed as an initial state common to all parametric calculations. These two simulations are carried out in compliance with the BCs described previously. On the figures provided afterwards, the time is given in seconds, as this is the time used in the numerical code (1 day = 86400 seconds). The time scheme for our creep simulations features an adaptive time step algorithm. In practice, in all the simulations carried out as part of this study, the entire simulation is performed over around 50 time steps.



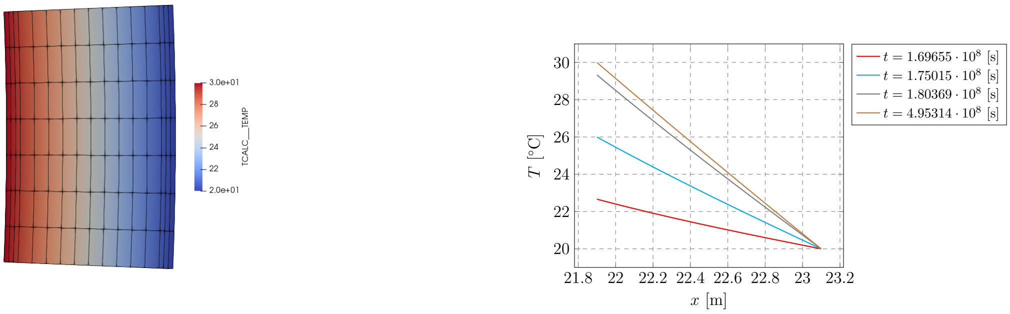
(a) View of the drying field C_w at the last time step of the HF simulation (top view).

(b) Evolution of the drying field C_w along x in the plane $(y = 0,\ z = 0)$.

Figure 4.17: Water content snapshots (output of the hydric calculation step) at the end of the HF simulation.

Figure 4.17 displays the water content in the standard section at the end of the HF calculation. This figure depicts the evolution of the C_w field in the thickness of the containment building (in the standard section). Likewise, Figure 4.18 shows the evolution of the temperature field in the thickness of the standard section. The physical parameters used for these calculations are summarized in Table 4.2 where undefined parameters are chosen as follows:

$$\begin{cases} \overline{\eta}_{\mathrm{dc}} &= 5 \cdot 10^{9}\ [\mathrm{Pa}^{-1}] \\ \overline{\kappa} &= 4.2 \cdot 10^{-4}\ [-] \\ \overline{\alpha}_{\mathrm{dc}} &= 7.56 \cdot 10^{-6}\ [-] \\ \overline{\eta}_{\mathrm{is}} &= 2.76 \cdot 10^{18}\ [\mathrm{Pa\,s}] \\ \overline{\eta}_{\mathrm{id}} &= 1.38 \cdot 10^{18}\ [\mathrm{Pa\,s}] \end{cases}$$

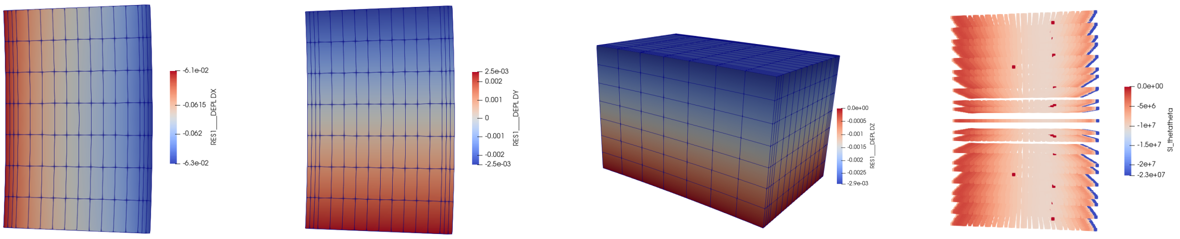


(a) View of the temperature field T at the last time step of the HF simulation (top view).

(b) Evolution of the temperature field T along x in the plane ($y = 0$, $z = 0$).

Figure 4.18: Temperature snapshots (output of the thermal calculation step) at the end of the HF simulation.

From these auxiliary fields (H field in the methodology formulation in Section 2.2) we can determine all the mechanical fields using the HF code. Figure 4.19 represents the displacement fields and the components of the Cauchy stress tensor obtained for the HF calculation we are seeking to reproduce in this section.



(a) u_r [m] (b) u_θ [m] (c) u_z [m] (d) $\sigma_{\theta\theta}$ [Pa]

Figure 4.19: Mechanical fields snapshots (displacements, cf. Figure 4.19a, 4.19b, 4.19c, and stresses within the concrete, cf. Figure 4.19d) at the end of the HF calculation on the standard section.

Our first goal is to ensure that the mechanical fields (displacements, stresses in the concrete and normal forces in the cables) are fairly accurate approximations of the values obtained from HF calculations. Besides, using a ROM of a standard section should provide a good quality approximation of the fields used in practical applications by engineers. In our case, this RSV has two main purposes: first, to compute leakage estimates from prestress loss in the cables, and second, to perform recalibration tests from deformation data (tangential and vertical deformations) on the intrados and extrados.

Figure 4.20 depicts the evolution of the mean value of the normal forces in each of the five cables within the standard section. For the record, the mesh studied contains two vertical cables and three horizontal cables. Within the framework of the investigated model, the vertical cables have a similar behavior (as do the three horizontal cables). In the following, we have decided to

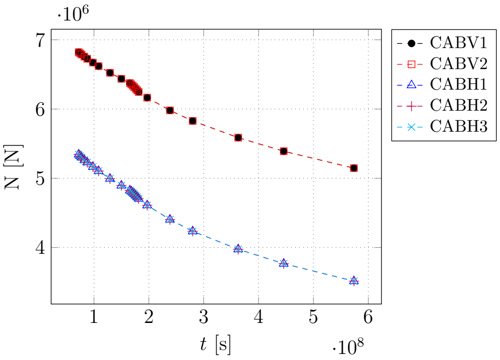
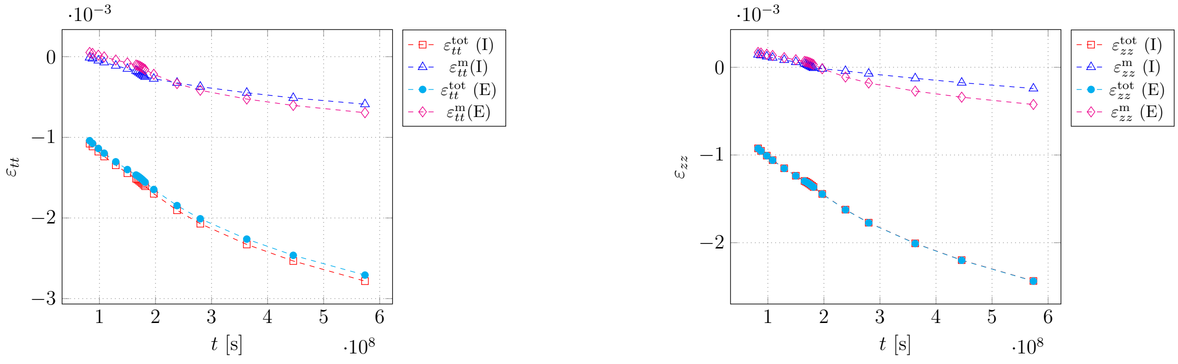


Figure 4.20: Evolution of normal forces in the two vertical (CABV1, CABV2) and three horizontal (CABH1, CABH2, CABH3) cables of the standard section.

report only the results for one horizontal and one vertical cables (CABV1 and CABH2), to ease the readability of the results. Figure 4.21 displays the evolution of mechanical strains and total strains in the concrete. In our notations, (I) stands for intrados whereas (E) stands for extrados. In our cases of interest, the total strains of the material are not purely mechanical. In general, data assimilation problems only focus on mechanical deformations. This is of key interest when reconstructing the strain field from the displacement modes, since the strain includes components due to temperature gradients and/or water pressure. Indeed, in our ROM resolution procedure, we have generalized coordinates at our disposal, which enable us to reconstruct the displacement field in the material. By computing the symmetric gradient of this displacement field, we can determine the total strains. In order to reconstruct a strain field, we must subtract the terms related to the thermal and hydric fields. Both these fields may be derived independently of the reduction process, since we only reduce the mechanical part of the calculation chain. Thus, we are able to pre-calculate the TH strain fields and subtract them from a total strain field so as to obtain the reconstructed mechanical strain field.



(a) ε_{tt}

(b) ε_{zz}

Figure 4.21: Comparison for the pointwise values between some components (tangential and vertical) the mechanical strains and the total strains in sensor zones (extrados (E) and intrados (I)).

In order to assess the accuracy of our reduced model, we introduce approximation errors for these different fields: for the average of the normal forces at the nodes in the CABV1 vertical cable ($E_{\overline{\mu}}^{\mathrm{app},(t)}[\mathrm{N}_{\mathrm{V}_2}]$), and in the horizontal cable ($E_{\overline{\mu}}^{\mathrm{app},(t)}[\mathrm{N}_{\mathrm{H}_2}]$), for the average of the tangential strain and vertical strain on the extrados ($E_{\overline{\mu}}^{\mathrm{app},(t)}[\varepsilon_{tt}^{\mathrm{m}}$ (avg - E)] and $E_{\overline{\mu}}^{\mathrm{app},(t)}[\varepsilon_{zz}^{\mathrm{m}}$ (avg - E)]) , and finally for the average of the tangential strain and horizontal strain on the intrados ($E_{\overline{\mu}}^{\mathrm{app},(t)}[\varepsilon_{tt}^{\mathrm{m}}$ (avg - I)] and $E_{\overline{\mu}}^{\mathrm{app},(t)}[\varepsilon_{zz}^{\mathrm{m}}$ (avg - I)]) . To average the components of the strain tensor, the values at the Gauss points are extrapolated to the nodes, and the value at the nodes is then averaged. These relative errors in the deformation fields relate exclusively to mechanical deformations. Indeed, this is the only part of the tensor that is actually modified by our reduction process, as explained above.

Speedups and approximation errors

In order to validate the ROM, we verify that the displacement field is properly reconstructed. Furthermore, since we are interested in the use of the ROM for engineering applications, it is necessary to confirm the quality of the approximation on the various quantities of interest, more precisely tangential and vertical deformations and normal forces in the cables (which enables us to calculate prestressing loss). Ultimately, it is crucial to provide a model that reduces the computation time required whenever a call is made. To this end, we focus on the speedups obtained after construction of the reduced model (online phase).

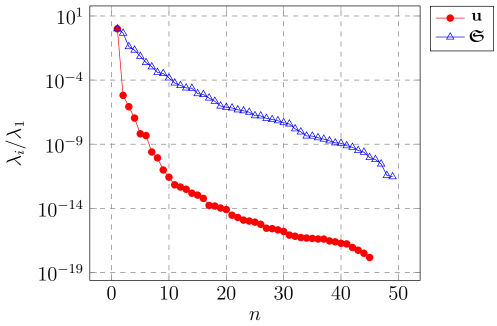
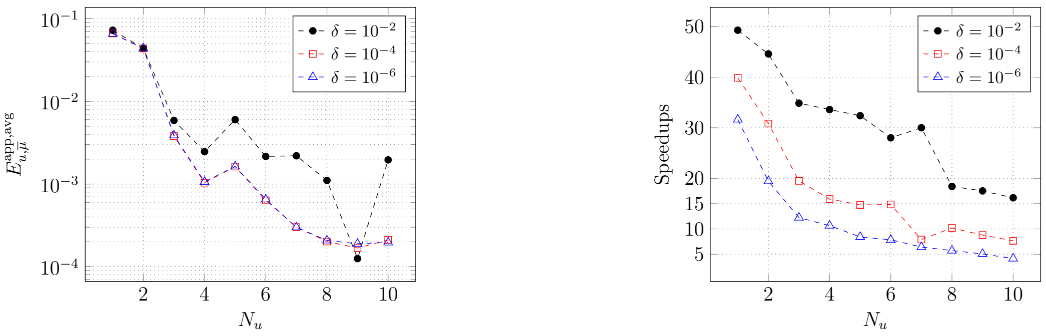


Figure 4.22: POD eigenvalues for the displacement ($\mathbf{u}$) and the generalized forces ($\boldsymbol{\mathfrak{S}}$) using a ℓ_2 compression for a solution reproduction problem (50 initial snapshots).

Figure 4.22 depicts the POD eigenvalues generated on snapshots of displacements ($\mathbf{u}$) and generalized forces ($\boldsymbol{\mathfrak{S}}$). The decay profiles are quite distinct between the two physical quantities: the decay of the eigenvalues for displacements is fast, unlike in the case of generalized forces. This implies that the sizes of the two bases generated for POD tolerances of the same order of magnitude are significantly different. The displacement basis will always be much smaller than the generalized force basis.

As a way of assessing the robustness of the reduction approach proposed here, we have built several ROMs for different numbers of displacement modes and different hyper-reduction tolerances. An increase in the number of modes and a decrease in the δ hyperparameter both improve the quality of the ROM and increase computation time (speedup). Thus, a tradeoff needs to be found for engineering applications in order to provide a fast and accurate ROM. Figure 4.23 displays the evolution of speedups and time-averaged displacement approximation errors as a function of the number of modes (for several tolerances). We observe that from 5 modes upwards, The

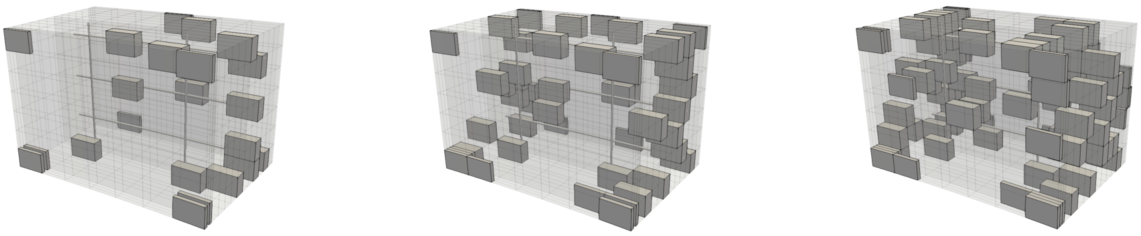


(a) Approximation errors on $\mathbf{u}$.

(b) Speedups.

Figure 4.23: Evolution of time-averaged approximation errors on the displacements and speedups as a function of the number of modes used (N_u, cf. Figure 4.23a) and for several hyper-reduction tolerances (δ, cf. Figure 4.23b).

reduced order model exhibits an good approximation quality, with approximation errors below the order of 0.2% (for all tolerances studied). In this case, the speedups achieved are substantial: around 10 for the most severe tolerance (equal to the Newton-Raphson tolerance), around 15 for the intermediate tolerance studied, and over 30 for the coarsest tolerance. These accelerations in CPU computation time are all the more appealing as the mesh studied in this study is very coarse, with only a few hundred elements (cf. Figure 4.24 for further details). This opens the door to future work on the use of finer meshes in NCB cross-section studies.



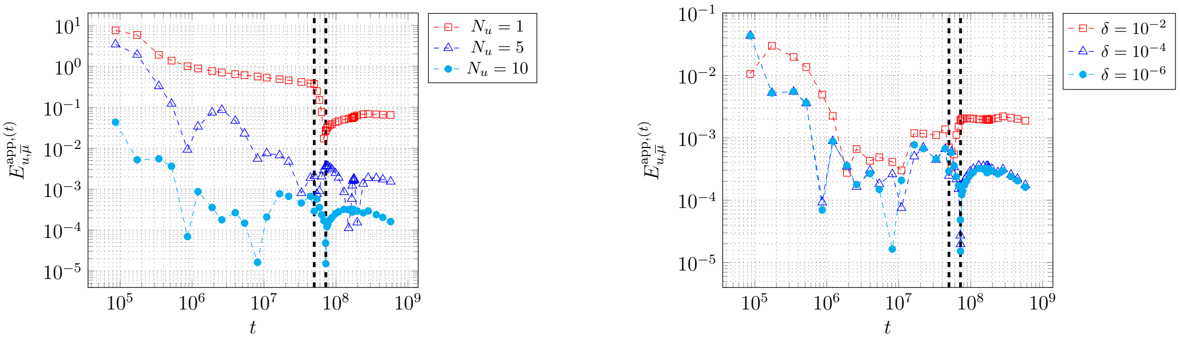
(a) $\delta = 10^{-2}$

(b) $\delta = 10^{-4}$

(c) $\delta = 10^{-6}$

Figure 4.24: Reduced meshes of the standard section obtained for a reproduction problem solution using $N_u = 5$ displacement modes and for several hyper-reduction parameters.

We have further investigated the quality of the ROM along the time trajectory of the problem. Figure 4.25 represents the relative errors at each time step for different ROMs. Since the construction of the ROM is determined by a pair of hyperparameters (N_u, δ), we focused on the influence of each parameter in fixing the second. The parameters set in the two test cases are chosen so as to be as restrictive as possible in the parameter sets we explore here. We find that for our problem, the number of modes has a much greater influence on time-evolution profiles than hyper-reduction tolerance. Since the latter parameter leads to an increase in mesh size as it decreases, this prompts us to state that: in this non-parametric case, it is advisable to fix a number of modes to control the approximation error, and it suffices to take a low or intermediate tolerance to get good speedups. We notice that for low approximation qualities, there are jumps in the relative error profiles of the displacement fields. This is due to the fact that the ROM is built over the entire life of the standard section, namely with three distinct physical regimes: life of concrete without cables, prestressing, and life of concrete with cables. For small numbers of modes, the ROMs is unable to generate modes designed to approximate these three phases. Since we chose to use no weighting, it will have a tendency to approximate the final step much more accurately, which is justified by the fact that the number of time steps associated with this phase is much greater. This higher approximation quality on the last step is of interest for our applications, as we seek not only a reliable approximation in terms of time trajectories, but also, and above all, a solution that is truly representative of the system's final state. If we need control the time-averaged approximation errors in a different manner, it would be natural to use a weighted POD in order to take into account the non-constant timestepping.



(a) Time evolution of relative errors for $\delta = 10^{-6}$ vectors in the reduced basis and varying number of N_u values.

(b) Time evolution of relative errors for $N_u = 10$ vectors in the reduced basis and varying number of δ values.

Figure 4.25: Evolution of approximation errors on displacements at each time step for several numbers of modes used or for several hyper-reduction tolerances.

Errors on the quantities of interest

The scope of the research we have undertaken requires us to be confident in our ability to provide accurate QoIs. Thus, we wish to verify that the ROM obtained, in addition to being a good approximation of the HF calculation in terms of displacements while being significantly less computationally expensive, can be used in real applications. This is achieved by investigating the profiles of normal forces in the cables and deformations at the sensor level (average measure of a component of the strains tensor over the internal or external face). We would like to point out that data post-processing differs according to the QoIs studied. The reduced mesh contains all the prestressing cables, while the quadrature laws are unchanged in the one-dimensional mesh. As a result, we can compute the relative error on normal forces directly after calling up the reduced model. For strains, however, we must reconstruct the strain fields on the HF mesh, and then apply the observation operators (physical sensors) used in the HF framework. This step s computationally inexpensive compared to the overall procedure, as the symmetric gradients of the modes are already known, because they are required for the hyper-reduction process. All that needs to be done is to multiply these modes to the generalized coordinates and apply the observation operator. Figure 4.26 provides the time-evolution of the relative errors on the QoIs. On Figure 4.26, we delimit the three phases of a mechanical calculation for a nuclear containment building: a first phase in which the cables are not involved in the mechanical calculation, i.e. the concrete evolves on its own; a second phase in which the concrete is prestressed (cf. Equation (3.14) for specific loads in this case); then, finally, the life of the prestressed concrete, in which the concrete and cables are fully coupled. The three periods are delimited by dotted black vertical lines. The HF solver's adaptive time-stepping process explains the temporal distribution of the various snapshots. The initial time for plotting corresponds to the first time step output by the reference calculation code.

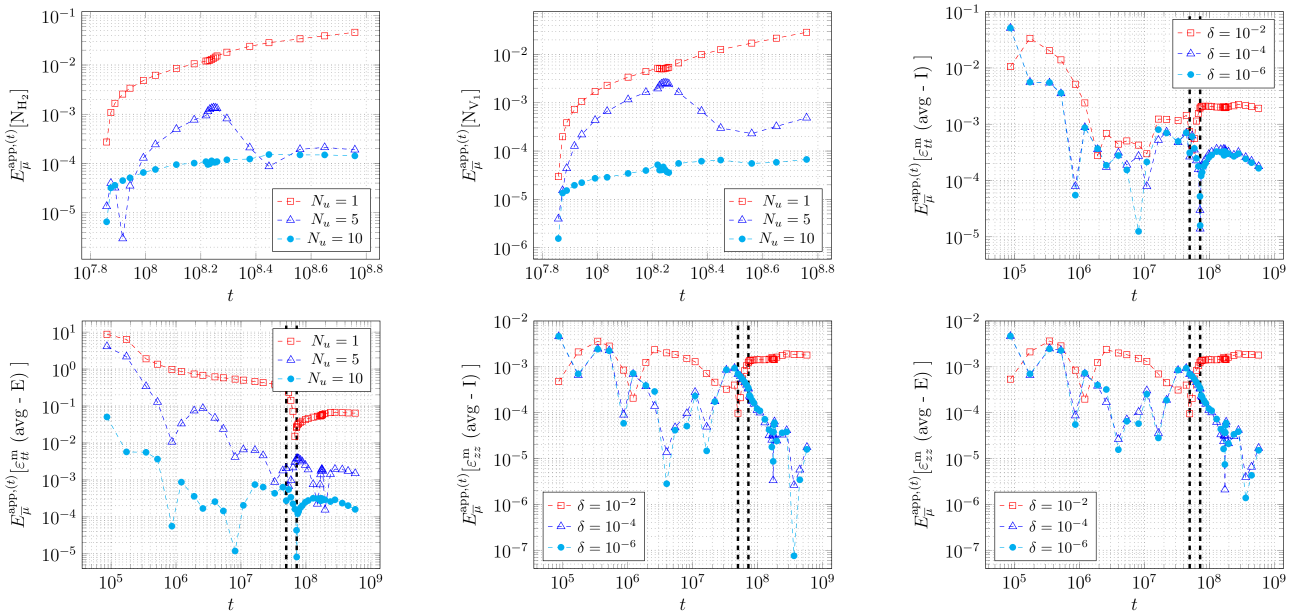


Figure 4.26: Evolution of approximation errors on QoIs at each time step for several numbers of modes used or for several hyper-reduction tolerances (the two vertical lines in black delimit the prestressing section of the cables).

The pattern of strain changes is similar to that of displacement approximation errors. Furthermore, the observation of a better approximation of deformations during the life of the NCB after prestressing is also confirmed. This confirms the usefulness of the ROM for data assimilation problems. In practice, data is only available once the cables have been prestressed. For the sake of clarity, we would like to point out that the time scale for the profile of relative errors in normal forces is not the same as that for deformations. In fact, only the life of the enclosure after prestressing is depicted, since normal forces are always zero beforehand, or known analytically.

4.3.3 Parametric problem

In a second step, we study a parametric case. As mentioned above, we consider here a strong-greedy approach. Thus, in order to drive the greedy search, we consider the maximum approximation error on a given training set (Θ_{train}), for the parameters we have not yet examined. As a reminder, Θ_* corresponds to the set of parameters used in building the ROM. We introduce a notation for the maximal error obtained when testing the ROM:

$$\Delta_N^{\text{stg}} = \max_{i\in\Theta_{\text{train}}\setminus\Theta_*} E_{u,\mu_i}^{\text{app,avg}}.$$

In the physical case under study, uncertainty is mainly limited to five physical parameters $\mu = [\eta_{\text{dc}}, \kappa, , \alpha_{\text{dc}}, \eta_{\text{is}}, \eta_{\text{id}}]^\top \in \mathbb{R}^5$, and in particular to the first two. As a validation of our model reduction approach, we set all the other parameters of the problem (see values in Table 4.2), and restrict the parametric problem to the other parameters.

Input parameter	**Notation**	**Value**	**Unit**
Young's modulus (steel)	E_{s}	$1.9 \cdot 10^{11}$	Pa
Poisson's ratio (steel)	ν_{s}	0.3	–
Density (steel)	ρ_{s}	7850	$\mathrm{kg\,m^3}$
Thermal dilation coefficient (steel)	$\alpha_{\mathrm{th,s}}$	$1 \cdot 10^{-5}$	K^{-1}
Guaranteed maximum load stress at break	f_{prg}	$1.86 \cdot 10^{9}$	Pa
Cable cross-section	S_{s}	$5400 \cdot 10^{-6}$	m
Young's modulus (concrete)	E_{c}	$4.2 \cdot 10^{10}$	Pa
Poisson's ratio (concrete)	ν_{c}	0.2	–
Density (concrete)	ρ_{c}	2350	$\mathrm{kg\,m^3}$
Thermal dilation coefficient (concrete)	$\alpha_{\mathrm{th,c}}$	$5.2 \cdot 10^{-6}$	K^{-1}
Autogenous shrinkage coefficient	β_{endo}	$66.1 \cdot 10^{-6}$	–
Dessication shrinkage coefficient	α_{dc}	**X**	–
Reversible deviatoric basic stiffness	k_{rd}	$5.98 \cdot 10^{18}$	Pa
Reversible deviatoric basic viscosity	η_{rd}	$8.12 \cdot 10^{16}$	Pa s
Irreversible deviatoric basic viscosity	η_{id}	**X**	Pa s
Basic creep activation energy	U_{bc}/R	4700	K
Basic creep reference temperature	T^0_{bc}	20	°C
Basic creep consolidation parameter	κ	**X**	–
Desiccation creep viscosity	η_{dc}	**X**	Pa^{-1}
Dead weight of upper concrete lifts	$\sigma_{z,c}$	$1.375 \cdot 10^{6}$	Pa
Stress applied to vertical cables	$\sigma_{v,s}$	$990.7 \cdot 10^{6}$	Pa
Stress applied to horizontal cables	$\sigma_{h,s}$	$1264.7 \cdot 10^{6}$	Pa

Table 4.2: Coefficients for the mechanical model fixed for the parametric problem. The notation **X** corresponds to the parameters that can vary and, therefore, we do not give *a priori* numerical values.

In-sample test for $\mathcal{P} \subset \mathbb{R}^2$

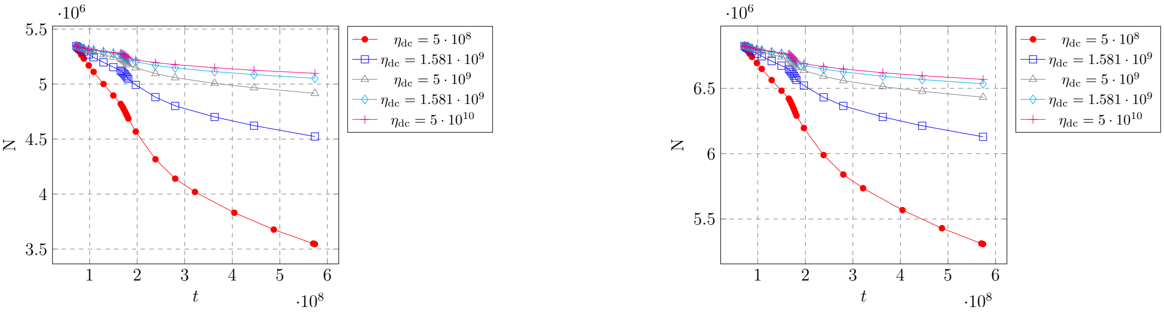
We confine the study to a parametric case with two parameters. The vector of parameters considered is as follows:

$$\mu = \begin{bmatrix} \eta_{\mathrm{dc}} \\ \kappa \end{bmatrix} \in \left[5 \cdot 10^8,\ 5 \cdot 10^{10}\right] \times \left[10^{-5},\ 10^{-3}\right] \subset \mathbb{R}^2.$$

This is tantamount to setting the following parameters (in addition to those given in Table 4.2):

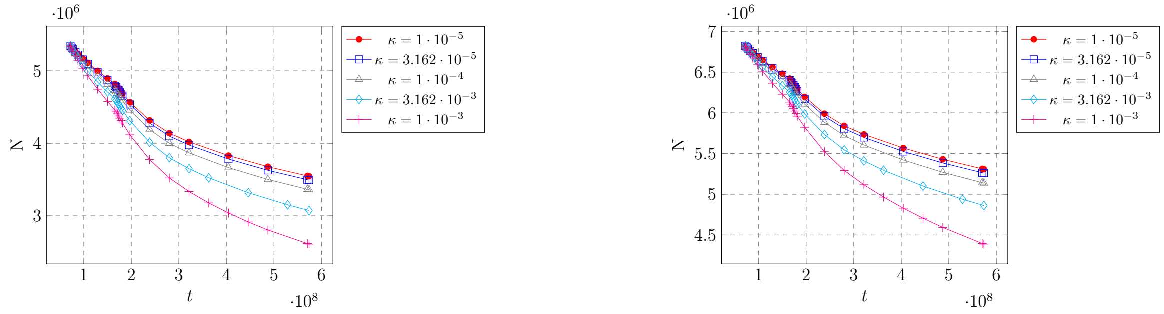
$$\overline{\alpha}_{\mathrm{dc}} = 7.56 \cdot 10^{-6}\ [-], \quad \overline{\eta}_{\mathrm{is}} = 2.76 \cdot 10^{18}\ [\mathrm{Pa\,s}], \quad \overline{\eta}_{\mathrm{id}} = 1.38 \cdot 10^{18}\ [\mathrm{Pa\,s}].$$

We rely on a training space of size $|\Theta_{\mathrm{train}}| = 25$, designed as the tensor product of two one-dimensional grids log-evenly spaced (5×5 grid). This choice results from a tradeoff between the need for sufficiently fine discretization to have several parameters, and the offline CPU cost of building the ROM (an HF calculation takes around fifteen minutes). The choice of optimal discretization is out of the scope of this work and is a field of research of its own. To help understand the physical problem under study, Figure 4.27 depicts the evolution of normal forces over time for different parameter sets. We can clearly appreciate that the loss of prestress in the cables (a key feature in the study of leakage rates) strongly differs according to the pair of parameters studied. The observation of these quantities supports the choice of a logarithmic discretization for the construction of the parametric grid.



(a) $\kappa = 1 \cdot 10^{-5}$ (horizontal cables)

(b) $\kappa = 1 \cdot 10^{-5}$ (vertical cables)



(c) $\eta_{\mathrm{dc}} = 5 \cdot 10^7$ (horizontal cables)

(d) $\eta_{\mathrm{dc}} = 5 \cdot 10^7$ (vertical cables)

Figure 4.27: Evolution of normal forces over time for pairs of parameters belonging to the parametric set of size $|\Theta_{\mathrm{train}}| = 25$. Figures 4.27a-4.27b (resp. Figure 4.27c-4.27d) feature cases where the parameter κ (resp. η_{dc}) is fixed. For each pair, we plot the time evolution of the normal forces averaged over all the nodes of the vertical and horizontal cables.

Figure 4.28 shows the decay of the POD eigenvalues when using the 25 HF snapshots. The decay is similar to that shown in Figure 4.22. We notice that for the parametric case, the decay is fast and the gain in compression will be significant.

As a first test, we report a quick evaluation of the construction of a ROM on a smaller training set, consisting of 4 points. In other words, we take only the extremums of the 2d square to which all the parameters belong. The aim of this simpler case is to compare the two methodologies for building POD-reduced bases (in the parametric case) before presenting the case on the 25-point parametric case. Figure 4.29 depicts the speedups and approximation errors obtained after 4 iterations (the maximum number of iterations possible for this case) for different pairs of hyper-parameters used for ROM construction: number of modes and hyper-reduction tolerance. We observe that the hierarchical basis strategy leads to an increase in basis size (in our case), which reduces speedup and improves approximation quality (to below one percent). On the other hand, the use of full POD enables much better speedups to be maintained, while reducing the approximation error, but to a lesser extent. The same tradeoff applies to ROM construction as described above. In the case studied here, the regularity of the problem (at least for this set of parameters), prompts us to favor a POD on all snapshots (therefore, the basis is not hierarchical

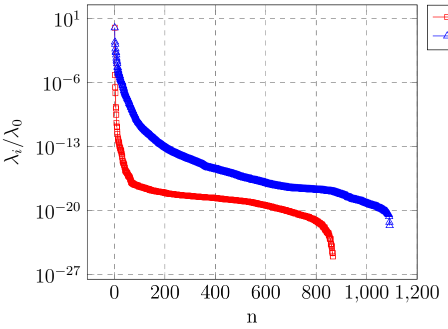
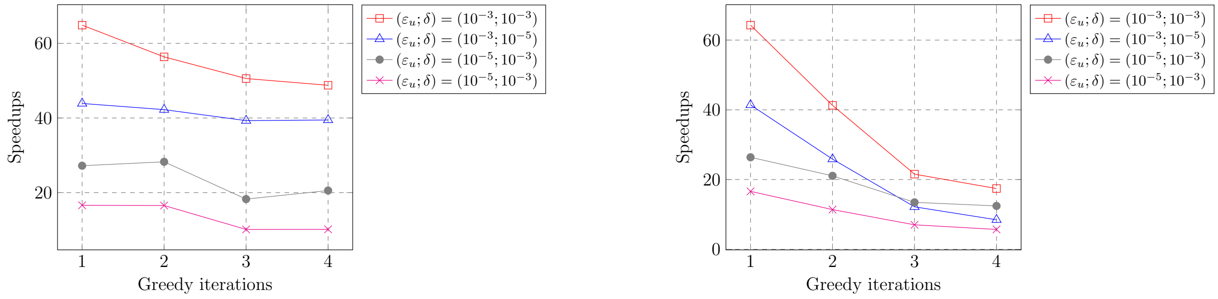


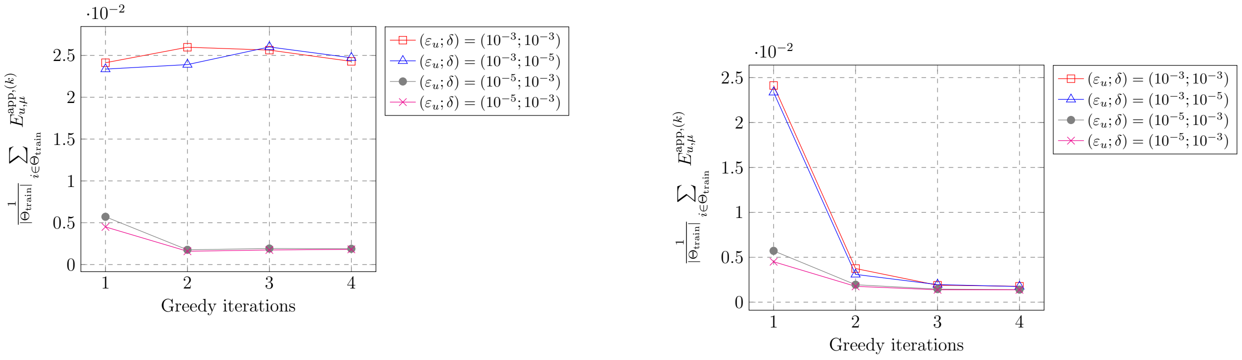
Figure 4.28: POD eigenvalues for the displacement and the generalized forces ($\mathfrak{S}$) using a ℓ_2 compression for a parametric problem.

during iterations), in order to have the most efficient ROM both in terms of computational gain, while having reasonable approximation errors.



(a) Speedups (POD on all HF snapshots).

(b) Speedups (Incremental POD).



(c) Average error (POD on all HF snapshots).

(d) Average error (Incremental POD).

Figure 4.29: Speedups and average approximation errors on displacements fields for $\mu \in \Theta_{\text{train}}$ using a training set of size $|\Theta_{\text{train}}| = 4$ for different compression tolerances (ε) and hyper-reduction parameters (δ) and comparison between non-incremental and incremental POD.

Then, we apply this strategy to a larger training set ($|\Theta_{\text{train}}| = 25$ parameters). Figure 4.30 represents the decay of the maximum approximation error on unexplored parameters (used to drive the greedy procedure). These successive choices clearly lead to a decrease in the maximum error (cf. Figure 4.32a) and the average error (cf. Figure 4.32b) over the entire training set (explored and unexplored parameters). Scaling up for each parameter, Figure 4.33 shows the time-averaged

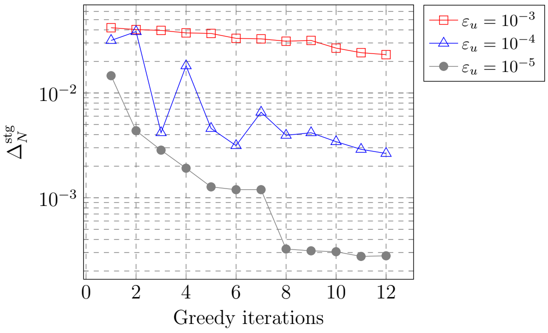


Figure 4.30: Maximum approximation error on unexplored parameters decreases during greedy iterations with an hyper-reduction parameter $\delta = 10^{-5}$.

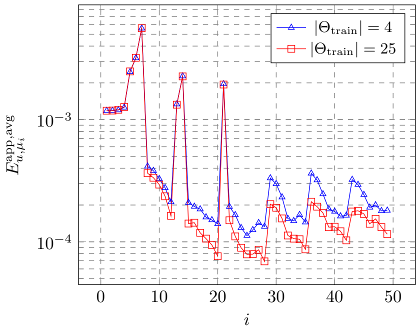


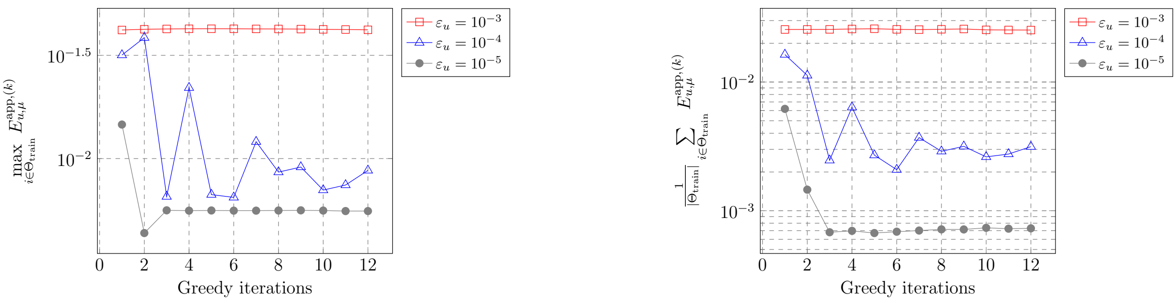
Figure 4.31: Statistical errors on the training set Θ_{train}, defined as a 5×5 grid along the greedy iterations. Two strategies are compared: POD on all HF snapshots (red), and incremental POD (orange).

approximation errors for each parameter over the first iterations of the algorithm. As confirmed by the other figures, we observe that for the case studied, we have errors of the order of a few percent on all parameters (no more than ten percent) after just a few iterations. This is due to the relative regularity of the problem studied. Figure 4.31 displays error statistics (median, quartiles) over the course of greedy iterations (5 by 5). We compare two approaches for incremental POD or POD on all snapshots, with error visualization, where we observe a decrease in medians over the iterations.

Out-of-sample test for $\mathcal{P} \subset \mathbb{R}^2$

All the above numerical results highlight the good approximation quality of the ROM on the training set. Nevertheless, it is crucial to further assess the methodology's suitability for out-of-sample parameters. To this end, we consider a 7-by-7 grid. This ensures that we get non-matching points. Then, we test the approximation quality of the ROM on this set, called the test set.

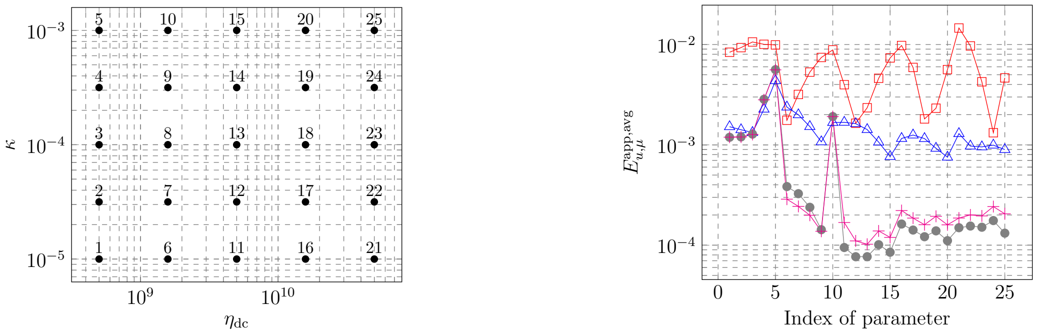
Figure 4.34 depicts boxplots for time-averaged approximation errors on the test set for the same training set for two sets of greedy strategies: one based on a POD on all snapshots (cf. Figure 4.34a) and the other on an incremental POD (cf. Figure 4.34b). From a statistical point of view, most of the test set features good approximation quality. The distribution of statistics across the two cases is consistent. For the POD on all snapshots, the error on the training set is of slightly higher quality than on the test set, while maintaining excellent approximation quality. Despite the



(a) Maximum error

(b) Average error

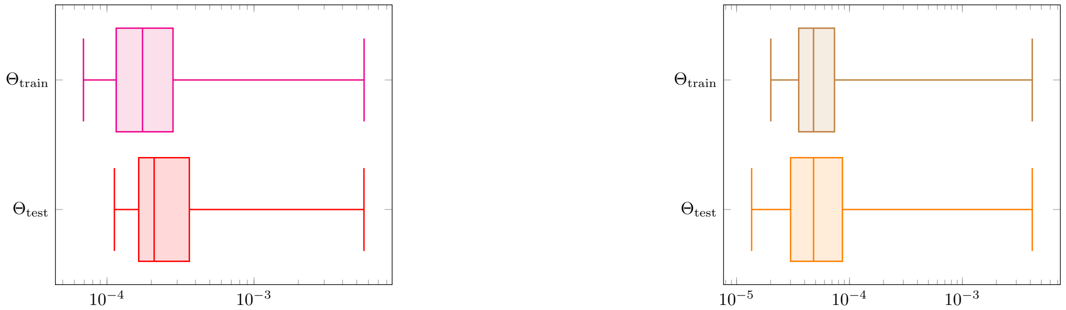
Figure 4.32: Average approximation errors on displacements fields for $\mu \in \Theta_{\text{train}}$ using a training set of size $|\Theta_{\text{train}}| = 25$ and a non-incremental POD for different compression tolerances (ε) with an hyper-reduction parameter $\delta = 10^{-5}$.



(a) Indexes of parameters

(b) Approximation errors

Figure 4.33: Time-averaged approximation errors on displacement on the training set ($|\Theta_{\text{train}}| = 25$) for the first greedy iterations with an hyper-reduction parameter $\delta = 10^{-5}$.

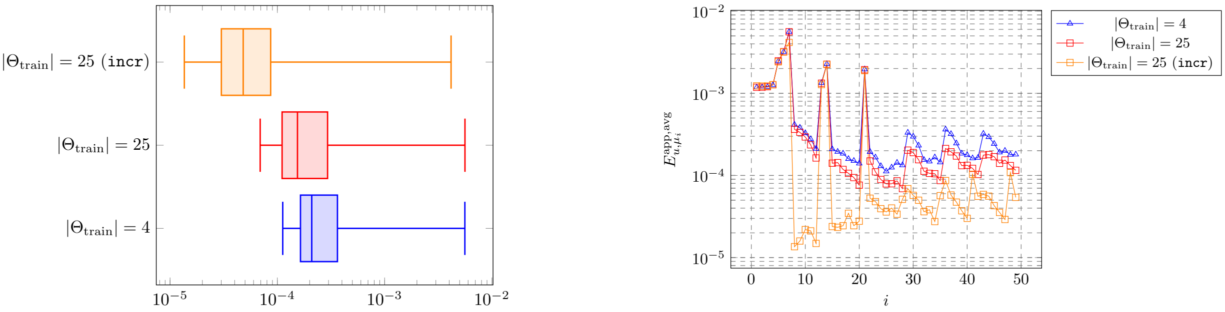


(a) POD on all HF snapshots.

(b) Incremental POD.

Figure 4.34: Boxplot for a training set on a 5×5 grid ($|\Theta_{\text{train}}| = 25$), verified on a test set on a 7×7 grid ($|\Theta_{\text{test}}| = 49$). The quantities measured are the time-averaged errors on each set, for a ROM resulting from a greedy procedure, stopped after 5 iterations.

simplicity of the case, it remains complex to perfectly capture the worst-case representations in the same way as the rest. Nevertheless, the worst-case error remains of the order of a few percent on the test set. For the case with incremental POD, the error quality between training and test sets is very similar, which is consistent with the fact that more modes are used than with POD on the snapshot set. Yet the difference between training and test sets is due to the smaller quartile spread on the training set (lower statistical dispersion), which is also coherent.



(a) Boxplots

(b) Approximation errors

Figure 4.35: Statiscal repartition of time-averaged errors generated by several ROMs on the same test set defined on a 7×7 grid ($|\Theta_{\text{test}}| = 49$). Three ROMs are compared (all obtained by a greedy process): built on a 2×2 training grid with POD on all HF snapshots (blue), on a 5×5 training grid with POD on all HF snapshots (red), and on a 5×5 training grid with an incremental POD (orange). Figure 4.35a is a boxplot of time-averaged errors on Θ_{test} and 4.35b is the time-averaged errors according to the number of the parameters in the Θ_{test} (numerotation is similar to Figure 4.33a, but on a 7×7 grid).

In a second step, we can also compare the greedy approaches with each other in terms of their behavior on the test set (cf. Figure 4.35). As can be expected, the poorest approximation case matches the case with the smallest training set size, followed by the case with 25 points and total POD, followed by a case with 25 points and incremental POD. This analysis is reflected in the boxplots (cf. Figure 4.35a), as well as in the plot of errors as a function of parameter indices (indices are distributed in a similar way to discretization on a 5x5 grid).

4.4 Conclusion

In the previous chapter, we outlined the various stages of a model reduction approach applied to quasi-static nonlinear mechanics problems in structural mechanics. The final targeted application concerns a standard section of a nuclear containment building. To this end, we presented two development stages, starting with the mono-modeling approach, followed by the multi-modeling approach. The reason for this division was to verify the generality of the proposed approach while offering a progression in the complexity of the problems tackled. A crucial element was to develop an approach compatible with the industrial-grade FE code, `code_aster`. The reduction methodology put forward is based on several principles, including the construction of a reduced model for nonlinear cases and a proposed construction approach. The first involves the construction of a reduced basis, the projection solver, as well as the creation of an empirical quadrature rule and the resulting reduced mesh. The second approach is greedy in nature, involving methods for building adapted reduced models, as well as the derivation of an error indicator in the case of mono-modeling situations.

For the validation of the single-modeling framework, we successfully implemented the method directly with an HF industrial code and validated it on an elastoplastic material. We proposed a time-averaged error indicator to drive the offline Greedy sampling, which is cost-efficient and has been shown numerically correlated to the approximation errors, and we developped an element-wise empirical quadrature procedure to reduce online costs. The whole procedure delivers impressive computational cost improvements in the order of $\mathcal{O}\left(20-25\right)$ with relative prediction errors in the order of 10^{-3}. For multi-modeling problems, we also provided ROMs designed to replicate the behavior of prestressed concrete with high speedups and good approximation errors. Those ROMs could open the door for their use in real engineering applications, since provide a good representation of the variables of interest used in practice by engineers, whether for structural state analysis (leakage rate study) or for in-depth data analysis (data-assimilation problem, Bayesian approaches). The next chapter proposes a numerical approach to pave the way for the use of these ROMs for calibration problems through data assimilation.

Nevertheless, the results achieved call for further research in several directions. Firstly, the meshes employed represent a simplified case of a standard section of containment buildings, with simplified loading (no consideration of ten-yearly visits). It might be of interest to investigate how the model reduction approach can be used to complexify the given problem for this same material behavior, i.e. the THM approach for prestressed concrete. This may include the adoption of a more refined, or more complex, mesh (part larger than the RSV, for instance a slice, or even a full mesh) to see if CPU gains and approximation quality are maintained in such cases. Another approach could be simply to consider time-varying loading, typically taking into account ten-yearly visits (vessel pressurization tests). A second point consists in increasing the size of the parametric space, since the test cases presented here are of small dimensions. Such tests would raise two questions in particular. First of all, does the proposed linear approach retain its efficiency (few modes in reduced spaces) as the size of the parameter space increases? Secondly, we might consider rendering the greedy process effective for large parameter spaces. In this respect, the extension and validation of the error indicators to the multi-modeling case remains an essential step towards a weak-greedy approach.

Chapter 5

A trust-region approach for parameter calibration through data-assimilation relying on-the-fly updated reduced order models

5.1 Introduction

Optimization problems guided by partial differential equations, also known as PDE-constrained optimization problems, natrually arise within diverse fields of engineering and science, and represent a significant research challenge [BGHvBW03][AKLR18]. Data assimilation (DA) problems fall notably within this class of problems, wherein the objective is to infer the characteristics of a system based on observed data and simulations. Our emphasis is on parameter calibration using data assimilation, which entails identifying the physical parameters that optimally represent the given data. Our particular focus is on a variational assimilation formulation. Within this framework, optimization relies on minimizing the discrepancy between data and model, while adhering to the constraint that the obtained solution aligns with the physics prescribed by the model. Despite the considerable advantages that widespread adoption of PDE-constrained optimization could bring to engineering and various scientific domains, the substantial computational costs in terms of time and resources, impedes its broad implementation.

This research primarily focuses on efficiently solving a DA problem, namely a specific class of optimization problems governed by PDEs in a deterministic setting. The resolution of this algorithmic category hinges on the application of minimization techniques, with gradient descent being a common class of algorithms. On the one hand, the study of PDE-constrained optimization problems can occur in the continuous domain, treating the PDE as a constraint which has to be met. In this case, optimality conditions form a system of partial differential equations that need to be analyzed and solved. Then, discretization occurs on this set of obtained equations. On the other hand, an alternative strategy involves directly working at the level of the discretized PDE and formulating the optimization problem at that level. This approach proves more practical for large-scale optimization problems or those relying on industrial codes or complex models. In such cases, the differentiation step is not problem-dependent, making it more suitable. This work exclusively considers the discrete formulation of the PDE-constrained optimization problem. However, as previously highlighted, such optimization problems can incur prohibitively high computational costs. The contribution of Alexandrov [ADJLT98][ALG$^+$01] establishes a practical and robust framework to hasten optimization problems, employing surrogate models to secure convergence towards a local optimum. This is achieved through a trust-region approach for resolving optimization problem. More precisely, this approach is applied for unconstrained optimization problems [ADJLT98] and nonlinearly constrained optimization [ALG$^+$01]. Moreover, the surrogate models employed for optimization acceleration can take various forms, such as response surfaces [FK09], projection-based ROMs [FS03][ZF15][QGVW17][ZCK19][YHZ21], or even deep learning [LMRC21]. In the context of this research, we rely on the use of projection-based ROMs using a trust-region approach for a specific category of deterministic PDE-constrained problem, namely DA problems.

In the first chapters, we focused on the offline/online paradigm for building ROMs. For the optimization procedure, our focus lies in harnessing the iterative nature of optimization processes to construct ROMs on-the-fly, thereby mitigating the computational burden. These methods are gaining popularity, offering a versatile approach to various optimization problems. For instance, the trust-region POD (TR-POD) method [AFS00][BC08], relying on the iterative construction of ROMs through POD, was developed for optimal control problems. These on-the-fly optimization methods extend their applications to other areas, including shape optimization [ZF15], topology optimization [YHZ21] and multi-scale parameter optimization [KO24]. Trust-region approaches have been particularly studied and tailored to ensure global convergence during iterations. Indeed, various studies have been conducted to integrate approximation theory related to ROMs with the aim of coupling it with optimization criteria. The incorporation of concepts such as error bounds and indicators has demonstrated its effectiveness, particularly in the context of linear problems [YM13][QGVW17]. Adaptations, including the development of error-awareness trust regions, have been tailored to address nonlinear PDEs, such as those encountered in subsonic aerodynamic shape optimization [ZF15]. The coupling of nonlinear problems with hyper-reduction, to the best of our knowledge, has a more limited body of literature concerning theoretical proofs of global convergence. However, recent research efforts have focused on extending such methods to establish globally convergent approaches for accelerating large-scale optimization [WZ23]. This methodology aims to provide globally convergent methods to expedite large-scale optimization processes.

This chapter unfolds in two primary segments. First, we embark on a comprehensive exploration of DA problems, aiming to define pertinent concepts and notations, and articulate the formulation of the optimization problem. Subsequently, we delve into the trust-region approach, a powerful optimization technique that can be effectively coupled with the utilization of ROMs. The subsequent discussion revolves around the synergy achieved through this coupling. The latter part of the chapter pivots towards a numerical analysis of various mechanical examples to validate the feasibility of employing this approach for our specific problems. When addressing the numerical solution of optimization problems, a conventional technique employed is the gradient method. Specifically, for constrained optimization problems, an effective strategy involves relying an adjoint method. To ease the testing of adjoint methods for gradient calculations, we first focus on elasticity problems, which we will introduce in Section 5.5 and 5.6. Such mechanical problems are characterized by a structure allowing for the analytical computation of sensitivities, namely, the derivatives of residuals with respect to parameters. These cases serve as ideal validation scenarios for the proposed method. In the final phase of our exploration, we extend the validation of the method to problems featuring internal variables and coupled with industrial codes. This endeavor solidifies the robustness and applicability of our approach across a spectrum of complex scenarios, marking a crucial step towards addressing the computational challenges inherent in DA processes. Indeed, iterations for optimization algorithms can number in the dozens. Regarding

CPU costs, simulating our target problem (standard section of a containment building) requires roughly ten minutes on available PCs, while the optimal industrial scenario — a full containment building— takes about 1 day for one HF simulation (for the mechanical step).

5.2 Data-assimilation problem formulation

5.2.1 Concept and notations

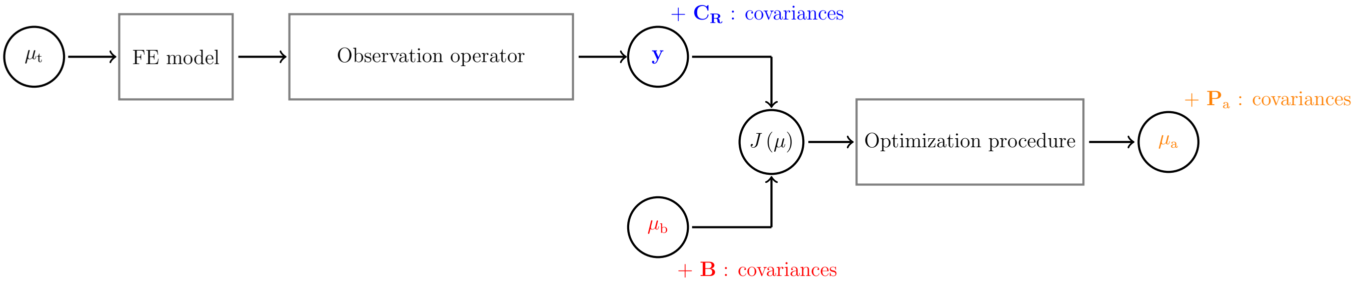


Figure 5.1: General principles of a DA problem (example with synthetic data generated from a true state μ_t). Observations are generated from a real physical parameter, with associated uncertainties. In addition, we have prior knowledge of the input parameter (μ_b), with an associated uncertainty. Optimization of the cost function ($J : \mathcal{P} \subset \mathbb{R}^p \to \mathbb{R}_+$) results in a set of parameters, known as the analysis (μ_a). It is also possible to obtain the analysis covariance matrix $\mathbf{P}_a$.

In this section, we present the classical concept of a DA problem [B$^+$14], and the associated notations (cf. Figure 5.1). In the context of the study of a numerical model (described by Equation (1.2) or (1.4) for example), we assume that the state of the system can be described by a discrete vector $\mu \in \mathcal{P} \subset \mathbb{R}^p$. Besides, we assume that there exist a representation μ_t of the true real state of our system. The objective of the approach is to provide a vector of parameters called analysis (μ_a) which is designed to combine an *a priori* knowledge of the system, embedded in a parameter called the background (μ_b), and observations of the physical system in its real behaviour ($\mathbf{y}$) to devise a reliable estimate of this reference true state μ_t. We define a map from the space of states ($\mathbb{R}^p$) with the space of observations ($\mathbb{R}^{n_{\text{obs}}}$), called the observation map H, which can be used to express the observation vector $\mathbf{y}$ thanks to the true state μ_t:

$$\mathbf{y} = H\left(\mu_t\right) + \mathbf{e}^0 \tag{5.1}$$

where $\mathbf{e}^0 \in \mathbb{R}^{n_{\text{obs}}}$ is the model error which aggregates both the instrumental errors (errors of the measurement tools, etc.) and the representation errors (representativeness of the physical model with respect to the real data). In order to model in greater detail these errors, we introduce the covariance matrix of the observation errors $\mathbf{C_R} \in \mathbb{R}^{n_{\text{obs}} \times n_{\text{obs}}}$ as follows[1]:

$$\left[\mathbf{C_R}\right]_{ij} = \mathbb{E}\left[\left[\mathbf{e}^0\right]_i, \left[\mathbf{e}^0\right]_j\right] \tag{5.2}$$

Likewise, we introduce the background error $\mathbf{e}^b \in \mathbb{R}^p$, which measures the deviation between an *a priori* estimate μ_b (best *a priori* knowledge before coupling with the data) and the true (unknown) state:

---

[1] Traditionally, in the literature, this covariance matrix is noted $\mathbf{R}$. However, this notation is used in our work for the residual, which explains the introduction of this new notation $\mathbf{C_R}$.

$$\mathbf{e}^{\mathrm{b}} = \mu_{\mathrm{b}} - \mu_{\mathrm{t}} \tag{5.3}$$

We introduce the covariance matrix of the background errors $\mathbf{B} \in \mathbb{R}^{p\times p}$:

$$[\mathbf{B}]_{ij} = \mathbb{E}\left[\left[\mathbf{e}^{\mathrm{b}}\right]_i, \left[\mathbf{e}^{\mathrm{b}}\right]_j\right] \tag{5.4}$$

These two matrices are assumed to be known to perform the DA problem. They are chosen so as to be positive definite (and therefore invertible). At last, we assume that the background and model errors are not correlated. When faced with a DA problem, the expert or engineer deliberately defines the level of uncertainty attributed to input measurements or their expertise (prior knowledge) to establish a deterministic approach. This method restricts distributions as it assumes Gaussian error distributions. However, this assumption enables the formulation of a deterministic optimization problem that can be resolved once the covariance matrices are provided.

5.2.2 Cost function for variational data-assimilation problem

As explained above, the DA process we present here belongs to the class of variational methods. This implies that we are addressing the minimization of a cost functional. The latter ($J^{\mathrm{hf}} : \mathcal{P} \to \mathbb{R}^{+}$) is defined as the sum of two terms, a first term that measures the deviation from the *a priori* knowledge (background) and a second term that measures the deviation from the measurements. The HF observation operator defined on the parameter set ($H^{\mathrm{hf}} : \mathcal{P} \to \mathbb{R}^{n_{\mathrm{obs}}}$), corresponds to a call to the HF solver and a post-processing to generate the observation vector. Thus, the HF cost function for the DA problem is expressed as follows:

$$J^{\mathrm{hf}}(\mu) = \frac{1}{2}\|\mu - \mu_{\mathrm{b}}\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\left\|\mathbf{y} - H^{\mathrm{hf}}(\mu)\right\|^2_{\mathbf{C}_{\mathbf{R}}^{-1}} \tag{5.5}$$

The problem regularization term is encapsulated in the first part of this cost function Much more, this term can be seen as the best state of knowledge the engineer can rely on, before using any data.

5.3 Trust-region approach for optimization problems

Following the DA framework considered within this work, we consider a constrained PDE optimization problem. Numerous numerical methods exist to solve this optimization problem. In this section, we focus on trust-region methods and, we provide details on such methodology for solving unconstrained optimization problems.

5.3.1 Trust-region approach for an optimization problem

We focus herein on optimization methods for unconstrained nonlinear problems, of the form:

$$\min_{\mu\in\mathbb{R}^p} \quad J(\mu) \tag{5.6}$$

where $J : \mathbb{R}^p \to \mathbb{R}_+$ is a real-valued function, which we call in our setting the HF cost function. Trust-region methods rely on an iterative process. Trust-region methods comprise a succession of cheaper sub-problems, only valid locally, so as to converge towards the optimal solution. The approximation model is constructed in conjunction with a function $\vartheta_k : \mathbb{R}^p \to \mathbb{R}_+$, called the constraint. This function is used in particular to define the trust region, $\mathcal{R}_k :=$

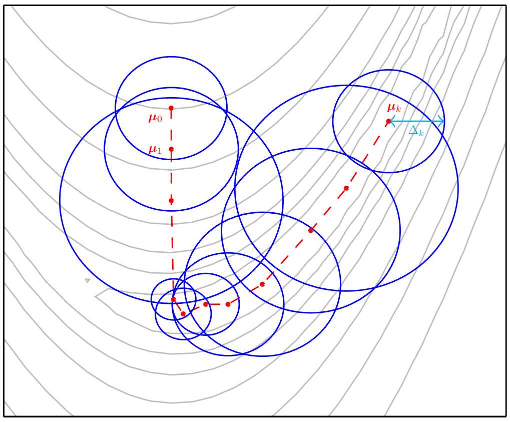


Figure 5.2: Principle of the trust-region approach with visualization of successive iterates and associated trust regions when solving an optimization problem.

$\{\mu \in \mathbb{R}^p, \quad \vartheta_k(\mu) \leq \Delta_k\}$, which matches the region where we solve the optimization sub-problem. Indeed, when the deviation between the approximation objective function and its HF counterpart is locally small at the center of the trust region μ_k, the quality of the approximation model is only assured locally. As a result, the optimization problem at the k-th iteration of the TR algorithm can be rewritten as follows:

$$\min_{\mu \in \mathcal{R}_k} \quad J_k(\mu) \tag{5.7}$$

In summary, using a trust-region approach involves creating a sequence of approximation models. We solve a sequence of optimization sub-problems, associated with a appropriate cost functions J_k and a given trust-region $\mathcal{R}_k$. As a consequence, we have a sequence of successive candidates μ_k (center of the trust regions) which will converge towards a solution, which we consider to be the solution to our global optimization problem. Figure 5.2 is a schematic diagram designed to illustrate the main ideas behind the method. The red curve defines the sequence of trust-region centers, seen as successive iterations of the global optimization problem. For each center, the resolution zone is defined by the blue areas. The subsequent candidate is determined by solving the approached problem within this specified zone.

TR approaches are particularly useful when the evaluation of the objective function or its gradient becomes cost-prohibitive. These methodologies offer an alternative to linesearch methods [Wri06][GMW19]. Since the objective is to reframe a problem for easier resolution, the approximation model (and its gradient) must be cheaper to evaluate than the HF objective function (and its gradient) if the methodology is to be appropriate. The assumptions adopted can modify the quality (or speed) of convergence of the methodology developed. However, the principle of the method remains the same. In the traditional framework of trust-region methods [CGT00], the model for the sub-problem is often taken as a quadratic approximation of the objective function J, with a constraint taken as a Euclidean distance from the center of the trust regions (an illustration of such a formulation is provided in Appendix C). In general, the approximation model is assumed to be twice continuously differentiable and the Hessians are assumed to be uniformly bounded (on $\mathbb{R}^p$, or at least on the set of $\mathcal{R}_k$ trust regions).

5.3.2 Details of the trust-region methodology for an optimization problem

Each iteration k of the proposed trust region method is divided into four main steps. First, we set up a trust region model, J_k, and the constraint ϑ_k. The choice of constraint will be discussed in more detail later. In a second step, we estimate a candidate point (μ_k^*) for the next iteration as a solution to the optimization problem given by Equation (5.7). There is no evidence that the candidate point obtained by solving the previous sub-problem effectively decreases the value of the objective function. It is then necessary to compare the value of the objective function evaluated at the candidate point, and to compare it with the value obtained for the center of the trust region. The actualization criterion is defined as follows:

$$\rho_k = \frac{J\left(\mu_k\right) - J\left(\mu_k^*\right)}{J_k\left(\mu_k\right) - J_k\left(\mu_k^*\right)} \tag{5.8}$$

Once the ratio is calculated, the quality of the resolution step (acceptance or non-acceptance of the candidate point) is assessed by comparing the value of the ratio obtained with 1. This unit value matches the case where the ratio between prediction and center is calculated for a perfect model $(J_k(\mu) = J(\mu))$. If the ratio is close to unity, the step is accepted and the trsut-region radius is increased. Otherwise (and even more so if the ratio is negative), the step is rejected and the radius of the trust-region is reduced. In practice, these conditions are managed using hyper-parameters to define different decision intervals: parameters governing candidate acceptance $(0 < \eta_1 < \eta_2 < 1)$, and parameters concerning radius updating $(0 < \gamma$ and $\Delta_{\max})$. Figure 5.3 displays a graphical summary of the insights to choose or not to choose the candidate and update the trust-region. We provide more details on the steps below.

If $\rho_k \leq \eta_1$, we consider that the model has not sufficiently decreased the HF objective function. Heuristically, it means that the size of the trust-region is too large in relation to the quality of the model used. In that case, the candidate is not accepted, and the radius of the trust-region (Δ_k) is chosen to decrease. The choice is made using the hyperparameter $0 < \gamma < 1$ as follows:

$$\Delta_{k+1} \leq \gamma \vartheta_k\left(\mu_k^*\right)$$

This option enables strict radius decay, while ensuring that the candidate point obtained no longer belongs to the trust-region. This ensures that the new candidate point is different, and that there is no infinite loop. This is referred to as an unsuccessful step. In the case where $\rho_k \in]\eta_1, \eta_2[$, the candidate is accepted, and the radius is not modified $\Delta_{k=1} = \Delta_k$. At last, if $\rho_k \geq \eta_2$, we speak of a very successful stage: the candidate is accepted and we increase the radius of the trust-region. Heuristically, we assume that the model is so good in the area under consideration that we explore a little further. In our case, the (arbitrary) choice is as follows:

$$\Delta_{k+1} = \min\left\{\frac{1}{\gamma}\Delta_k,\ \Delta_{\max}\right\}$$

In practice, computing the evaluation criterion (cf. Equation (5.8)) requires calls to the HF solver, thereby incurring additional computational costs. Approaches exist in the literature to employ approximation functions during this step to reduce the computational expense.

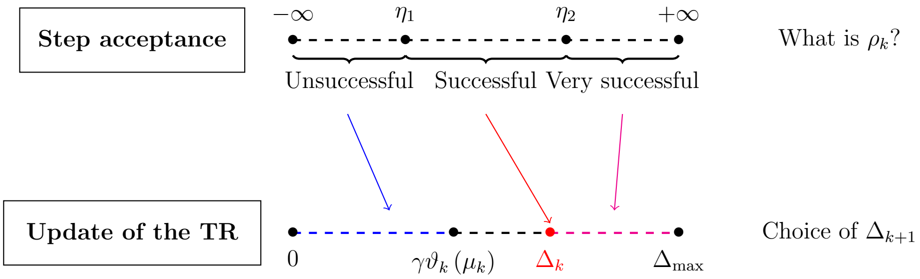


Figure 5.3: Step acceptance and update of the trust-region.

Algorithm 3 Trust-region method with exact evaluation of the objective function.

```
1: Initialisation: Choice of the initial parameters
2: Update of the constraint and the models: Formulation of the sub-problem so that global
   convergence is satisfied. Choose:
     • a model $J_k : \mu \in \mathbb{R}^p \rightarrow J_k(\mu) \in \mathbb{R}$
     • a constraint $\vartheta_k : \mu \in \mathbb{R}^p \rightarrow \vartheta_k(\mu) \in \mathbb{R}$,
3: Candidate: Approximate solution of the sub-problem to obtain a candidate $\mu_k^* \in \mathbb{R}^p$:
       $\min_{\mu \in \mathcal{R}_k} J_k(\mu)$
   where the candidate verifies Cauchy's fractional decay condition.
4: Actual-to-predicted decrease ratio: Compute performance criterion $\rho_k$
5: Step assessment and radius update:
     • if $\rho_k \geq \eta_1$, then we use the candidate $\mu_{k+1} = \mu_k^*$,
     • if not, do not change the center of the trust-region center $\mu_{k+1} = \mu_k$.
6: Update of the trust-region:
     • if $\rho_k \leq \eta_1$, then $\Delta_{k+1} \in ]0, \gamma\vartheta_k(\mu_k^*)]$,
     • if $\eta_1 < \rho_k < \eta_2$, then $\Delta_{k+1} \in ]\gamma\vartheta_k(\mu_k^*), \Delta_k]$,
     • if $\rho_k \geq \eta_2$, then $\Delta_{k+1} \in [\Delta_k, \Delta_{\max}]$.
```

5.4 Trust-region approaches with on-the-fly model order reduction for static problems

In this section, we introduce the methodology for addressing a static problem (cf. Equation (1.2)), which serves as an initial stage in tackling an inverse problem. It is noteworthy that this approach can be readily expanded to quasi-static scenarios, with emphasis on the system's ultimate state, for instance. Throughout the numerical experiments conducted, we will elaborate on potential extensions or adjustments of the method tailored for quasi-static cases.

5.4.1 On the use of reduced order models as approximation models

As mentioned above, the objective of our work consists in using ROMs to address a DA problem. This implies that we assume that the approximation models studied are ROM-based. Within our framework, the ROM is built on-the-fly and thus, the reduced order basis may be updated throughout the procedure. Indeed, as specified previously, we adapt the ROM at every new trust-region center μ_k, and, therefore, change the reduced order basis. To this end, we denote by $\mathbf{Z}_{u,k}$ the reduced order basis at the DA iteration k. We denote by $N_{u,k}$ the number of modes at the DA iteration k. We also define the reduced discrete residual at DA iteration k, denoted $\mathbf{R}_k^{\mathrm{r}} : \mathbb{R}^{N_u} \times \mathbb{R}^p \to \mathbb{R}^{N_u}$, as a function of the generalized coordinates and the parameter. This reduced residual is determined as before by projecting the reduced basis onto the HF residual $\mathbf{R}^{\mathrm{hf}} : \mathbb{R}^{N_u} \times \mathbb{R}^p \to \mathbb{R}^{N_u}$:

$$\forall \left(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \boldsymbol{\mu}\right) \in \mathbb{R}^{N_u} \times \mathbb{R}^p, \qquad \mathbf{R}_k^{\mathrm{r}} \left(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \boldsymbol{\mu}\right) = \mathbf{Z}_{u,k}^{\top} \mathbf{R}^{\mathrm{hf}} \left(\mathbf{Z}_{u,k} \widehat{\boldsymbol{\alpha}}_{u,\mu},\ \boldsymbol{\mu}\right) \tag{5.9}$$

Since we possess a sequence of ROMs, we should redefine a reduced cost function at each iteration. To this end, we introduce a cost function related to the reduced problem ($J_k^{\mathrm{r}} : \mathcal{P} \to \mathbb{R}^+$). The difference between this functional and the HF functional is the call to the reduced observation operator ($H_k^{\mathrm{r}} : \mathcal{P} \to \mathbb{R}^{n_{\mathrm{obs}}}$), i.e the call to the reduced solver. In comparison with the HF functional (cf. Equation (5.5)), the first term is unchanged, and only the observation operator differs in the second term:

$$J_k^{\mathrm{r}} (\mu) = \frac{1}{2} \left\|\mu - \mu_{\mathrm{b}}\right\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \left\|\mathbf{y} - H_k^{\mathrm{r}} (\mu)\right\|_{\mathbf{C}_{\mathbf{R}}^{-1}}^2 \tag{5.10}$$

Given the previous definitions of the HF and reduced cost functions through $F(\cdot, \cdot)$ defined in Equation (5.12), the cost functions can be expressed as follows:

$$\begin{cases} J^{\mathrm{hf}} (\mu) = F\left(\mathbf{u}_\mu,\ \mu\right), & \text{where} \quad \mathbf{u}_\mu = \mathcal{S}^{\mathrm{hf}} (\mu) \\ J_k^{\mathrm{r}} (\mu) = F_k^{\mathrm{r}} \left(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu\right) = F\left(\widehat{\mathbf{u}}_\mu,\ \mu\right), & \text{where} \quad \widehat{\mathbf{u}}_\mu = \mathbf{Z}_{u,k} \widehat{\boldsymbol{\alpha}}_{u,\mu} = \mathcal{S}_k^{\mathrm{r}} (\mu) \end{cases}$$

The difference between this functional and the HF functional is the call to the reduced observation operator ($H_k^{\mathrm{r}} : \mathcal{P} \to \mathbb{R}^{n_{\mathrm{obs}}}$), i.e the call to the reduced solver. In comparison with the previous functional (cf. Equation (5.5)), the first term is unchanged, and only the observation operator differs in the second term:

$$J_k^{\mathrm{r}} (\mu) = \frac{1}{2} \left\|\mu - \mu_{\mathrm{b}}\right\|_{\mathbf{B}^{-1}}^2 + \frac{1}{2} \left\|\mathbf{y} - H_k^{\mathrm{r}} (\mu)\right\|_{\mathbf{C}_{\mathbf{R}}^{-1}}^2 \tag{5.11}$$

In the foregoing steps, the problem is formulated so as to consider observation operators whose input data are the parameters. However, within this work, we focus on the investigation of calibration problems through DA in solid mechanics. Hence, the observation vector concerns the displacement field or a derived field. Data may directly pertain to the displacement field, or deformations (symmetric gradient), or velocities/accelerations for dynamical problems. This assumption is not restrictive and is consistent with experimental practice. Indeed, for static or quasi-static solid mechanics problems (focus of our work), real data generally concern displacements, and often even strains (image correlation with speckle, strain gauge on a sample, for example), since stress measurements are hard to come by. On the basis of this assumption, the observation operator $H : \mathbb{R}^{\mathcal{N}} \to \mathbb{R}^{n_{\mathrm{obs}}}$ obtained from the numerical simulation can be expressed in terms of the displacement field only. For greater clarity on the links between the various operators introduced, the operators on the parameters may be expressed using this single observation operator on displacements and the FE solver:

$$\begin{cases} H^{\mathrm{hf}} = H \circ \mathcal{S}^{\mathrm{hf}}(\mu) \quad \text{where} & \begin{array}{cccc} \mathcal{S}^{\mathrm{hf}} & : & \mathcal{P} & \rightarrow & \mathbb{R}^{\mathcal{N}} \\ & & \mu & \mapsto & \mathbf{u}_{\mu} \end{array} \\ \\ H^{\mathrm{r}}_k = H \circ \mathcal{S}^{\mathrm{r}}_k(\mu) \quad \text{where} & \begin{array}{cccc} \mathcal{S}^{\mathrm{r}}_k & : & \mathcal{P} & \rightarrow & \mathbb{R}^{\mathcal{N}} \\ & & \mu & \mapsto & \mathbf{Z}_{u,k}\widehat{\boldsymbol{\alpha}}_{u,\mu} \end{array} \end{cases}$$

where the functional $\mathcal{S}^{\mathrm{hf}}$ (resp. $\mathcal{S}^{\mathrm{r}}_k$) corresponds to the call to the HF (resp. reduced) solver. This remark helps us to recast this optimization problem as a PDE-constrained optimization problem, where the cost function is the same in both cases, but where the use of a ROM transforms the PDE constraint. More specifically, we introduce the cost function taking as inputs a discrete displacement field $\mathbf{u}$ and a parameter μ, defined as follows:

$$F(\mathbf{u},\ \mu) = \frac{1}{2}\|\mu - \mu_{\mathrm{b}}\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\|\mathbf{y} - H(\mathbf{u})\|^2_{\mathbf{C}_{\mathbf{R}}^{-1}} \tag{5.12}$$

This formulation results in a cost function that is solver-agnostic, assisting in the derivation of the adjoint method. Such a definition divides the said function into two terms, one depending exclusively on the parameter ($F^{\mathrm{b}} : \mathcal{P} \rightarrow \mathbb{R}^+$) and the other on the displacement ($F^{\mathrm{o}} : \mathbb{R}^{\mathcal{N}} \rightarrow \mathbb{R}^+$):

$$F(\mathbf{u},\ \mu) = F^{\mathrm{b}}(\mu) + F^{\mathrm{o}}(\mathbf{u}), \quad \text{where} \quad \begin{cases} F^{\mathrm{b}}(\mu) & = & \frac{1}{2}\|\mu - \mu_{\mathrm{b}}\|^2_{\mathbf{B}^{-1}} \\ F^{\mathrm{o}}(\mathbf{u}) & = & \frac{1}{2}\|\mathbf{y} - H(\mathbf{u})\|^2_{\mathbf{C}_{\mathbf{R}}^{-1}} \end{cases}$$

5.4.2 Penalization for dealing with the trust-region constraint for a PDE-constrained optimization problem

Once the optimization problem is set, the next step is to come up with a suitable resolution strategy for the trust-region sub-problem. We wish to solve the DA problem for a given ROM, in the trust-region $\mu \in \mathcal{R}_k$. When focusing on the problem formulation, this amounts to adding a constraint to the reduced optimization problem given by Equation (5.19):

$$\begin{array}{cl} \min\limits_{\mu \in \mathbb{R}^p} & J^{\mathrm{r}}_k(\mu) = F^{\mathrm{r}}_k(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu) \\ \text{subject to} & \begin{cases} \mathbf{R}^{\mathrm{r}}_k(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \mu) & = & 0 \\ \vartheta(\mu) & \leq & \Delta_k \end{cases} \end{array} \tag{5.13}$$

Our adopted strategy aims to maintain the PDE constraint as an equality constraint, simplifying the problem to focus solely on this remaining consideration. We decide to adopt a penalty-based approach, reformulating the problem using a logarithmic barrier [N+18, p. 56]. Thus, the principle is to come back to a problem of the form:

$$\begin{array}{cl} \min\limits_{\mu \in \mathbb{R}^p} & \phi^{\gamma^{\mathrm{r}}}_k \\ \text{subject to} & \mathbf{R}^{\mathrm{r}}_k(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \mu) = 0 \end{array}$$

where $\phi^{\gamma^{\mathrm{r}}}_k$ may be expressed explicitly as a function of the cost function $J^{\mathrm{r}}_k(\mu)$. In this case, we can go back to the previous situation by adapting the gradient calculation. The function $\phi^{\gamma^{\mathrm{r}}}_k$ depends on a hyper-parameter γ^{r}. As the latter approaches 0, the function $\phi^{\gamma^{\mathrm{r}}}_k$ converges towards the cost function in the reduced case (J^{r}_k). In practice, two approaches are available, according to the conditioning of the problem. A direct approach is to keep to a single parameter (and therefore a single function to be minimized) for the logarithmic barrier. Nevertheless, many iterations may be required for convergence. If successive systems are too ill-conditioned, it is more appropriate to employ a sequence of penalty parameters which converges towards the desired penalty parameter. As the sequence tends towards the desired parameter, the solution from the previous iteration is used as the initialization point for the new sub-problem. Within this work, we consider the following logarithmic barrier:

$$\phi_k^{\gamma^{\mathrm{r}}}(\mu) = \begin{cases} J_k^{\mathrm{r}}(\mu) - \gamma^{\mathrm{r}} \log\left[\Delta_k - \vartheta_k(\mu)\right] & \text{if} \quad \Delta_k - \vartheta_k(\mu) > 0 \\ +\infty & \text{otherwise} \end{cases} \tag{5.14}$$

5.4.3 On-the-fly model reduction strategy through optimization iterations

As mentioned before, the reduced basis is updated when a new center of the trust-region is obtained. We introduce a set $\mathbb{U}_k^{\mathrm{train}}$, which corresponds to the set of snapshots used for the construction of the reduced basis at iteration k. A first approach consists in incrementally enriching the training set, by adding the last obtained center. In a second step, a POD is performed to obtain the reduced basis.

$$\mathbb{U}_k^{\mathrm{train}} = \mathbb{U}_{k-1}^{\mathrm{train}} \cup \{\mathbf{u}_{\mu_k}\}, \quad \text{then} \quad \mathbf{Z}_{u,k} = \mathrm{POD}\left\{\mathbb{U}_k^{\mathrm{train}},\ \varepsilon_{\mathrm{POD},u}\right\}$$

Then, it remains to define an initial set for the construction of the reduced basis. A natural view is to restrict to the first trust-region center, which in our setting is the background:

$$\mathbb{U}_0^{\mathrm{train}} = \{\mathbf{u}_{\mu_{\mathrm{b}}}\}$$

Indeed, the background is the best knowledge of the parameters of the mechanical problem. It therefore seems consistent to use it as the initial point of our problem.

5.4.4 Choice of the constraint for the trust-region in the case of reduced order models

As specified in the algorithm, a key step remains the definition of a constraint to set the trust-region. This affects the shape of the trust-regions, and, thus, convergence. In this work, we consider two types of trust region. On the one hand, we consider a classical trust-region, i.e. using a Euclidean distance. In order to avoid being biased by differences in order of magnitude (adimensionalization), one solution may be to use a matrix norm $\mathbf{A}$, which aims to normalize the parameters. On the other hand, we consider an error-aware trust-region, inspired by the work of [Zah16]. To this end, we consider a constraint defined on the grounds of an a posteriori error indicator:

$$\vartheta_k(\mu) = \begin{cases} \|\mu - \mu_k\|_A, & \text{for an appropriate } \mathbf{A} \in \mathbb{R}^{p \times p} & (\texttt{'classical'}) \\ \Delta_{N,\mu}, & \text{as an error indicator} & (\texttt{'errind'}) \end{cases} \tag{5.15}$$

5.4.5 Gradient computation for the data-assimilation problem using reduced order models

Several methodologies can be used to solve this optimization problem. We provide here a gradient-based methodology. Such an approach requires computing the gradients of the cost functions (with respect to the parameter). An adjoint approach for PDE constrained optimization [WNLDZ92][Jam88] is employed to derive the gradient that can be then used for the optimization process. First, we provide an explanation of the adjoint approach for the HF problem (some additionnal details on the derivation are provided in Appendix D). This involves formulating a set of equations, known as the adjoint equations, which specifically concern a variable $\boldsymbol{\Lambda}$, known as the adjoint or dual variable. The primary objective is to ascertain these variables to subsequently compute a gradient estimate based on their values.

Adjoint approach for the high-fidelity problem

As mentioned above, we start by describing the adjoint approach in the case of an HF calculation. Thus, we focus on the following constrained optimization problem:

$$\begin{aligned} \min_{\mu \in \mathbb{R}^p} \quad & J^{\mathrm{hf}}(\mu) = F(\mathbf{u}_\mu,\ \mu) \\ \text{subject to} \quad & \mathbf{R}^{\mathrm{hf}}(\mathbf{u}_\mu,\ \mu) = 0 \end{aligned} \tag{5.16}$$

Therfore, the gradient of the cost function for a HF problem is calculated as follows:

$$\nabla J^{\mathrm{hf}}(\mu) = \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) - \boldsymbol{\Lambda}^\top \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu), \tag{5.17}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{\mathcal{N}}$ is solution of the adjoint equation:

$$\left[\frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]^\top \boldsymbol{\Lambda} = \left[\frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]^\top \tag{5.18}$$

where $\frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ and $\left[\frac{\partial F^{\mathrm{hf}}}{\partial \mathbf{u}}\right]^\top \in \mathbb{R}^{\mathcal{N}}$.The previous formulation provides an adjoint method of gradient calculation without specifying the expression of the cost function. Nevertheless, the cost function for a DA problem has a general expression given by Equation (5.12), with a splitting in two terms, each depending on $\mathbf{u}$ or μ. The cost function can then be derived explicitly with respect to displacement and parameter:

$$\begin{cases} \dfrac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) = \dfrac{\partial F^{\mathrm{b}}}{\partial \mu}(\mathbf{u}_\mu, \mu) = \mathbf{B}^{-1}\left(\mu - \mu^{\mathrm{b}}\right) \\ \dfrac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu) = \dfrac{\partial F^{\mathrm{o}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu) = -\left[\mathbf{H}(\mathbf{u}_\mu)\right]^\top \mathbf{C}_{\mathbf{R}}^{-1}\left(\mathbf{y} - H(\mathbf{u}_\mu)\right) \end{cases}$$

where $\mathbf{H}$ is the tangent operator of the observation operator, which includes the model (whether HF or reduced):

$$\left[\mathbf{H}(\mathbf{u}_\mu)\right]_{ij} = \frac{\partial H_i}{\partial \mu_j}(\mathbf{u}_\mu)$$

Numerical strategy for the adjoint approach for the reduced problem

This point opens a discussion on the choice of methodology for the appropriate computation of the gradient in the case of a reduced problem. For the HF case, the methodology is quite clear and was explained previously, that it to say that we rely on an adjoint method for gradient computation. For the case of a reduced models, two choices are possible. A first method may consist of relying solely on the adjoint and reducing this problem in order to have a reduced adjoint problem. In

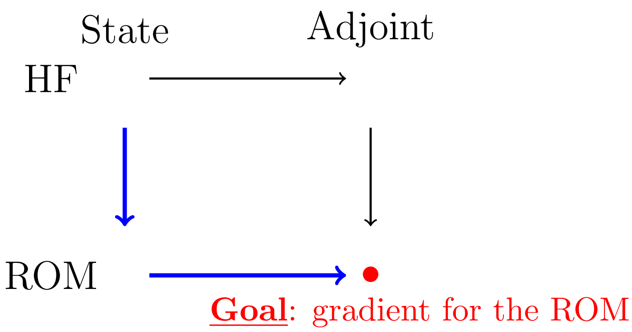


Figure 5.4: Choice of the methodology for the gradient computation in the case of a ROM problem. The **bold blue** arrows illustrate the path followed to derive the adjoint for the reduced model that will be used in a gradient method. The reduction is only performed on the state and then the adjoint is computed from the reduced residual. The arrows in **black** depict an alternative path that could have been considered for gradient derivation using an adjoint method.

this case, we have two reduced problems, one associated with the state and a second associated with the adjoint. This raises a question about the appropriate (coupled) reduction of these two problems. A second approach is to derive the adjoint only by knowing the given reduced problem, meaning that we only reduce the state, and from that we derive the adjoint. A priori, this method is less robust, but this allows us to only have a single reduced model, solely linked to the state. In this work, we opt for this choice. The reader may rely in part on the illustration given in the Figure 5.4 for a better understanding of our choices. This also explains the presentation of the methodology in the following sections. We present the calculation of the adjoint given a numerical model. Therefore, this calculation holds whether the initial model considered is the HF or reduced calculation. First, we present the gradient calculation by an adjoint method for the HF calculation, then that obtained for the ROM (given a reduction only on the state). We can also extend these remarks to previous choices. In fact, within our methodology, we adopt a *discretize-then-differentiate* approach [Gun02]. This means that all the choices (and available in Figure 5.4) are made starting from the discrete problem. Another approach could consist of deriving directly at the continuous level (derivation of the adjoint equation) before discretizing the problem. This is not the choice made throughout this work. The definition of our numerical strategy can also be interpreted in terms of error control. Indeed, as recalled above, our choice is as follows: we proceed from the discrete formulation, we construct a reduced approximation model, and we derive an adjoint from the ROM constructed on the state. This means that we have error control on the quality of the discrete HF model, on the construction of the ROM on the state, with an error control on the reduced model using error indicator or control in approximation error, but not on the adjoint.

Derivation of the adjoint approach for the reduced problem

In this following paragraph, we focus solely on the calculation of the gradient and, therefore, the adjoint in the reduced case. As a reminder (cf. Figure 5.4), we start from the constrained optimization problem with a reduction on the state:

$$\begin{aligned} \min_{\mu \in \mathbb{R}^p} \quad & J_k^{\mathrm{r}}(\mu) = F_k^{\mathrm{r}}(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu) \\ \text{subject to} \quad & \mathbf{R}_k^{\mathrm{r}}(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \mu) = 0 \end{aligned} \tag{5.19}$$

where all operators appearing in the formulation are defined in Section 5.2.2. We can repeat the same process as for the HF calculation to obtain the gradient of the cost function. The gradient of the cost function is now expressed as such:

$$\nabla J_k^{\mathrm{r}}(\mu) = \frac{\partial F_k^{\mathrm{r}}}{\partial \mu}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu\right) - \left[\boldsymbol{\Lambda}_k^{\mathrm{r}}\right]^\top \frac{\partial \mathbf{R}_k^{\mathrm{r}}}{\partial \mu}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu\right), \tag{5.20}$$

where $\boldsymbol{\Lambda}_k^{\mathrm{r}} \in \mathbb{R}^{N_{u,k}}$ is solution of the following adjoint equation:

$$\left[\frac{\partial \mathbf{R}_k^{\mathrm{r}}}{\partial \boldsymbol{\alpha}_{u,\mu}}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu\right)\right]^\top \boldsymbol{\Lambda}_k^{\mathrm{r}} = \left[\frac{\partial F_k^{\mathrm{r}}}{\partial \boldsymbol{\alpha}_{u,\mu}}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu},\ \mu\right)\right]^\top \tag{5.21}$$

We rewrite the previous set of equations in terms of the HF operators initially introduced. To this end, we draw on the relationship between the generalized coordinates and the total displacement fields when using a ROM. By relying on Equations (5.9) and (5.4.1), and by applying a chain rule, we derive the derivatives of the operators with respect to the generalized coordinates:

$$\left\{\begin{array}{ccccccc}
\dfrac{\partial F_k^{\mathrm{r}}}{\partial \boldsymbol{\alpha}_{u,\mu}} & = & \dfrac{\partial F_k^{\mathrm{r}}}{\partial \mathbf{u}}\dfrac{\partial \mathbf{u}}{\partial \boldsymbol{\alpha}_{u,\mu}} & = & \dfrac{\partial F_k^{\mathrm{r}}}{\partial \mathbf{u}}\mathbf{Z}_{u,k} & = & \dfrac{\partial F}{\partial \mathbf{u}}\mathbf{Z}_{u,k} \\[2ex]
\dfrac{\partial \mathbf{R}_k^{\mathrm{r}}}{\partial \boldsymbol{\alpha}_{u,\mu}} & = & \dfrac{\partial \mathbf{R}_k^{\mathrm{r}}}{\partial \mathbf{u}}\dfrac{\partial \mathbf{u}}{\partial \boldsymbol{\alpha}_{u,\mu}} & = & \dfrac{\partial \mathbf{R}_k^{\mathrm{r}}}{\partial \mathbf{u}}\mathbf{Z}_{u} & = & \mathbf{Z}_{u,k}^\top\dfrac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\mathbf{Z}_{u,k}
\end{array}\right.$$

Finally, following the same steps as in the HF case (cf. Equations (5.17) and (5.18)), the gradient of the cost function is expressed in the reduced case as:

$$\nabla J_k^{\mathrm{r}}(\mu) = \frac{\partial F_k^{\mathrm{r}}}{\partial \mu}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \mu\right) - \left[\boldsymbol{\Lambda}_k^{\mathrm{r}}\right]^\top \mathbf{Z}_{u,k}^\top \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu}(\underbrace{\mathbf{Z}_{u,k}\widehat{\boldsymbol{\alpha}}_{u,\mu}}_{=\widehat{\mathbf{u}}_\mu}, \mu) = \mathbf{B}^{-1}\left(\mu - \mu^{\mathrm{b}}\right) - \left[\boldsymbol{\Lambda}_k^{\mathrm{r}}\right]^\top \mathbf{Z}_{u,k}^\top \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu}\left(\widehat{\mathbf{u}}_\mu, \mu\right) \tag{5.22}$$

where $\boldsymbol{\Lambda}_k^{\mathrm{r}} \in \mathbb{R}^{N_{u,k}}$ is solution of the following adjoint equation:

$$\left[\mathbf{Z}_{u,k}^\top \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\left(\widehat{\mathbf{u}}_\mu, \mu\right)\mathbf{Z}_{u,k}\right]^\top \boldsymbol{\Lambda}_k^{\mathrm{r}} = \mathbf{Z}_{u,k}^\top \left[\frac{\partial F}{\partial \mathbf{u}}\left(\widehat{\mathbf{u}}_\mu, \mu\right)\right]^\top \tag{5.23}$$

Derivation of the logarithmic barrier

In the framework of the solver for the trust-region sub-problem, we rely on the logarithmic barrier defined in Equation (5.14) in order to take the trust-region into account. Thus, we need to compute the gradient of the logarithmic barrier defined in Equation (5.14). Such a development requires the computation of the gradient of the reduced cost function (see Section 5.4.5) and the computation of the constraint. The gradient of the logarithmic barrier is expressed as follows:

$$\begin{aligned}
\nabla \phi_k^{\gamma^{\mathrm{r}}}(\mu) &= \nabla J_k^{\mathrm{r}}(\mu) + \frac{\gamma^{\mathrm{r}}}{\Delta_k - \vartheta_k(\mu)}\nabla \vartheta_k(\mu) \\
&= \mathbf{B}^{-1}\left(\mu - \mu^{\mathrm{b}}\right) - \left[\boldsymbol{\Lambda}_k^{\mathrm{r}}\right]^\top \mathbf{Z}_{u,k}^\top \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu}\left(\widehat{\mathbf{u}}_\mu, \mu\right) + \frac{\gamma^{\mathrm{r}}}{\Delta_k - \vartheta_k(\mu)}\nabla \vartheta_k(\mu)
\end{aligned}$$

5.5 Numerical analysis of a data-assimilation problem for a linear elastic problem

In this section, we introduce a linear elasticity test case. This simple case provides an easily replicable case whose gain in terms of speedups for the ROM does not require a hyper-reduction process. Hence, it is possible to easily extend the number of tests in order to display the interests of the method.

First, we present the physical case used, i.e the constitutive equation, geometry and loadings, as well as the framework of the DA problem (choice of the formulation of the uncertainties and observation operator). In a second step, we apply the methodology without using trust-regions (which also matches the case with an infinite trust-region radius, $\Delta_k = \infty$). Third, we demonstrate the good approximation quality obtained for the case of DA with trust-regions. Bearing in mind that all these results are given for gradient approaches, and for example quasi-Newton approaches (more precisely BFGS [Fle00]), for the sub-problem resolution, we test alternatives solvers for the successive sub-problems, in particular using approximate gradient or gradient-free methods. The robustness of the method when changing the optimization solver ensures that we have an approach that can be generalized to cases where the gradients are difficult to extract (in the industrial code at stake, namely `code_aster`) or where we face difficulty obtaining sensitivities.

We present numerical results, with examples built using the open-source `Fenics` software [ABH$^+$15][LL17], which has proven its relevance to solid mechanics simulation [Ble18]. `Fenics` is used in this Chapter as the HF code for numerical simulation of mechanical problems. Additional modules have been added to build ROMs from the generated snapshots, with the development of associated TR solver. To this end, we relied on the numerical solvers available in `Python` and in particular `scipy` [VGO$^+$20].

5.5.1 Problem formulation for a linear elastic problem

Physical problem formulation

We consider a static linear elastic material. Thus, we focus on small-displacement small-strains problems. The equilibrium equation, the constitutive equations and the boundary conditions used are summarized through the following set of equations:

$$\left\{\begin{array}{rcll} -\nabla \cdot \sigma_\mu & = & f_v & \text{on } \Omega \\ \sigma_\mu & = & \mathcal{F}^\sigma_\mu\left(u_\mu\right) & \\ u_\mu & = & u_\mathrm{d} & \text{on } \Gamma_\mathrm{d} \\ \sigma_\mu \cdot n & = & f_s & \text{on } \Gamma_\mathrm{n} \end{array}\right. \tag{5.24}$$

where u_μ is the displacement field, σ_μ is the Cauchy stress tensor, $\partial\Omega = \Gamma_\mathrm{d} \cup \Gamma_\mathrm{n}$, and $\mathcal{F}^\sigma_\mu$ stands for the constitutive equation. The latter in linear elasticity can be expressed as:

$$\sigma_\mu = \mathcal{F}^\sigma_\mu\left(u_\mu\right) = \lambda \mathrm{tr}\left(\varepsilon\left(u_\mu\right)\right) 1 + 2G\varepsilon\left(u_\mu\right)$$

where G is the shear modulus and λ the first Lamé's parameter and the symmetric gradient of the displacement is defined as $\varepsilon\left(\cdot\right) = \nabla_s \cdot = \frac{1}{2}(\nabla \cdot + \nabla \cdot^\top)$. The variational formulation of the static problem defined by Equation (5.24) is expressed as follows: find $u_\mu \in \mathcal{X}_\mathrm{bc}$ such that $\forall v \in \mathcal{X}_\mathrm{bc}$:

$$\mathcal{R}\left(u_\mu, v, \mu\right) = \int_\Omega \lambda\left(\nabla \cdot u_\mu\right)\left(\nabla \cdot v\right) + 2G\nabla_s u_\mu : \nabla_s v \; dx = \int_\Omega f_v v \; dx + \int_{\Gamma_\mathrm{n}} f_s v \; ds,$$

where $\mathcal{X}_{\mathrm{bc}} := \{v \in \mathcal{X} : \ u = u_{\mathrm{d}} \text{ on } \Gamma_{\mathrm{d}}\}$. The pair of parameters (G, λ) can be rewritten relying on the Young's modulus E and the Poisson's ratio ν:

$$G = \frac{E}{2\,(1+\nu)} \quad \text{and} \quad \lambda = \frac{E\nu}{(1+\nu)\,(1-2\nu)}$$

We define the parameters for the parametric problem as follows:

$$\mu = \begin{bmatrix} E \\ \nu \end{bmatrix} \in \mathbb{R}^2 \tag{5.25}$$

We apply a problem discretization with a continuous Galerkin FE method. We state the variational form of the problem at stake using the FE space as the approximation space: find $u_\mu \in \mathcal{X}^{\mathrm{hf}}_{\mathrm{bc}}$ such that $\forall v \in \mathcal{X}^{\mathrm{hf}}_{\mathrm{bc}}$:

$$\mathcal{R}^{\mathrm{hf}}\left(u_\mu, v, \mu\right) = \int_\Omega \lambda \left(\nabla \cdot u_\mu\right)\left(\nabla \cdot v\right) + 2G \nabla_s u_\mu : \nabla_s v \ dx = \int_\Omega f_v v \ dx + \int_{\Gamma_{\mathrm{n}}} f_s v \ ds,$$

Such a formulation allows use to express the HF discrete residual with terms with are $\mathbf{u}$-dependent and others which are μ-dependent:

$$\text{find } \mathbf{u} \in \mathbb{R}^{\mathcal{N}}, \quad \text{such that} \quad \mathbf{R}^{\mathrm{hf}}\left(\mathbf{u},\ \mu\right) = \mathbf{A}\left(\mu\right)\mathbf{u} - \mathbf{b},$$

with the affine decomposition of the stiffness matrix:

$$\mathbf{A}\left(\mu\right) = \frac{E}{1+\nu}\mathbf{A}_1 + \frac{E\nu}{(1+\nu)\,(1-2\nu)}\mathbf{A}_2, \quad \text{where} \quad \left\{ \begin{array}{lcl} \mathbf{v}^\top \mathbf{A}_1 \mathbf{u} & = & \displaystyle\int_\Omega \left(\nabla \cdot \mathbf{u}\right)\left(\nabla \cdot \mathbf{v}\right) \ dx \\ \mathbf{v}^\top \mathbf{A}_2 \mathbf{u} & = & \displaystyle\int_\Omega \nabla_s \mathbf{u} : \nabla_s \mathbf{v} \ dx \end{array} \right.$$

These residual expressions are of great interest, since they enable several operators to be computed efficiently. From the standpoint of the reduction process, the reduced operators are easily computable, and the affine decomposition means we can reduce the computational cost of ROM calls. From a DA perspective, such a decomposition provides an analytical calculation of sensitivities, required when applying an adjoint method.

Reduced operators

As mentioned before, the given problem has a parametrically affine decomposition. It is then convenient to precompute the reduced operators, once the basis is known. To this end, we define the reduced discrete residual as follows:

$$\begin{aligned} \mathbf{R}^{\mathrm{r}}\left(\widehat{\boldsymbol{\alpha}}_{u,\mu}, \mu\right) &= \mathbf{Z}_u^\top \mathbf{R}^{\mathrm{hf}}\left(\widehat{\mathbf{u}}_\mu, \mu\right) \\ &= \frac{E}{1+\nu}\mathbf{Z}_u^\top \mathbf{A}_1 \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{u,\mu} + \frac{E\nu}{(1+\nu)\,(1-2\nu)}\mathbf{Z}_u^\top \mathbf{A}_2 \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{u,\mu} - \mathbf{Z}_u^\top \mathbf{b} \\ &= \frac{E}{1+\nu}\mathbf{A}_1^{\mathrm{r}} \widehat{\boldsymbol{\alpha}}_{u,\mu} + \frac{E\nu}{(1+\nu)\,(1-2\nu)}\mathbf{A}_2^{\mathrm{r}} \widehat{\boldsymbol{\alpha}}_{u,\mu} - \mathbf{b}^{\mathrm{r}} \end{aligned}$$

The matrices $\mathbf{A}_1^{\mathrm{r}}$ and $\mathbf{A}_2^{\mathrm{r}}$, and the vectors $\mathbf{b}^{\mathrm{r}}$ can be computed once the reduced basis is known. At this point, only an evaluation (and a sum of matrices) is required when calling the ROM. To estimate the quality of the ROM's approximation to the HF model, a number of a posteriori error indicators have been developed. In this work, we introduce an *a posteriori* error indicator defined as:

$$\Delta_{N_u,\mu} = \left\|\mathbf{R}^{\mathrm{hf}}\left(\widehat{\mathbf{u}}_{\mu},\ \mu\right)\right\|_{*}$$

where $\|.\|_{*}$ is an appropriate norm chosen depending on the norm used for the POD compression.

5.5.2 Setup of the test cases for twin experiments

We aim to scale the quantities in the cost functional in order to better set out the optimization problem at hand. In this section, we set out to establish a clear testbed for formulating the problem to be solved. The objective is to provide a set of hyper-parameters that enables us to define a range of problems, common in assimilation. We seek to evaluate if the method aligns with our requirements. Rather than employing real data, we conduct twin experiments, generating synthetic data from the numerical model. The data is generated from a parameter by means of the HF code, and is used (with or without noise) before carrying out a conventional optimization process. The focus lies in retracing the physical parameter μ_{t} used in generating the data. Thus, our objective is to introduce a method and validate its robustness, so that it can subsequently be applied to real problems. Therefore, we need to specify a scope where we are able to deal with several types of problem, but sufficiently narrow and framed to avoid confusion and unnecessary multiplication of calculations. To introduce our proposed formulation, we consider the case where the observation vector is a scalar. The scalar formulation is intended only to help us understand the underlying philosophy. We introduce the following notations:

$$E_{\mathrm{x}} = E - E_{\mathrm{b}},\ \nu_{\mathrm{x}} = \nu - \nu_{\mathrm{b}},\ y_{\mathrm{x}} = y - H\left(\mu\right)$$

Then, we can rewrite the HF cost function as:

$$J^{\mathrm{hf}}\left(\mu\right) = \frac{E_{\mathrm{x}}^2}{\sigma_E^2} + \frac{\nu_{\mathrm{x}}^2}{\sigma_\nu^2} + \frac{y_{\mathrm{x}}^2}{\sigma_{\mathrm{obs}}^2}$$

We introduce quantities related to the dimension of each physical quantity (λ_E and λ_ν for the parameters, and λ_{obs} for the observations) and its order of magnitude. Furthermore, we define correlation quantities based on the dimensionless physical quantities ($\overline{\sigma}_{\mathrm{obs}}$, $\overline{\sigma}_{\mathrm{obs}}$, and $\overline{\sigma}_{\mathrm{obs}}$). We can now express a cost functional as follows:

$$\begin{aligned}\sigma_{\mathrm{obs}}^2 J^{\mathrm{hf}}\left(\mu\right) &= \left(\frac{\sigma_{\mathrm{obs}}}{\sigma_E}\right)^2 E_{\mathrm{x}}^2 + \left(\frac{\sigma_{\mathrm{obs}}}{\sigma_\nu}\right)^2 \nu_{\mathrm{x}}^2 + y_{\mathrm{x}}^2 \\ &= \left(\frac{\overline{\sigma}_{\mathrm{obs}}}{\overline{\sigma}_E}\right)^2 \left(\frac{\lambda_{\mathrm{obs}}}{\lambda_E}\right)^2 E_{\mathrm{x}}^2 + \left(\frac{\overline{\sigma}_{\mathrm{obs}}}{\overline{\sigma}_\nu}\right)^2 \left(\frac{\lambda_{\mathrm{obs}}}{\lambda_\nu}\right)^2 \nu_{\mathrm{x}}^2 + y_{\mathrm{x}}^2\end{aligned}$$

We introduce the following notations, which refer to the observations:

$$\begin{cases} \phi_E &= \dfrac{\overline{\sigma}_E}{\overline{\sigma}_{\mathrm{obs}}} \\ \phi_\nu &= \dfrac{\overline{\sigma}_\nu}{\overline{\sigma}_{\mathrm{obs}}} \end{cases} \quad \text{and} \quad \begin{cases} \Lambda_E &= \dfrac{\lambda_E}{\lambda_{\mathrm{obs}}} \\ \Lambda_\nu &= \dfrac{\lambda_\nu}{\lambda_{\mathrm{obs}}} \end{cases}$$

We choose to treat the part related to observations as an identity matrix for the purposes of these studies. Consequently, it suffices to introduce these ratios associated with the units to achieve a dimensionless approach. Moreover, it is sufficient to control the ratios between the correlations by considering an identity correlation matrix for the observations, in order to control the weighting between the observation and background terms. In practice, we consider the inverses of the correlation matrices :

$$\mathbf{C}_{\mathbf{R}}^{-1} = \mathbf{I} \quad \text{and} \quad \mathbf{B}^{-1} = \begin{bmatrix} \frac{1}{\Lambda_E^2 \phi_E^2} & 0 \\ 0 & \frac{1}{\Lambda_\nu^2 \phi_\nu^2} \end{bmatrix}$$

As a test of the robustness of the developed algorithm, we shall consider a deviation between the background parameter and the parameter used for the twin experiment. The discrepancy between μ_t and μ_b is adjusted so as to have a relative *a priori* error between these two parameters ($\varepsilon_\mu^{\mathrm{err}}$).

$$\mu_\mathrm{t} = \mu_\mathrm{b} + \varepsilon_\mu^{\mathrm{err}} = \begin{bmatrix} E_\mathrm{b} \pm \overline{\sigma}_{\mathrm{err},E} E_\mathrm{b} \\ \nu_\mathrm{b} \pm \overline{\sigma}_{\mathrm{err},\nu} \nu_\mathrm{b} \end{bmatrix} = \begin{bmatrix} \left(1 \pm \overline{\sigma}_{\mathrm{err},E}\right) E_\mathrm{b} \\ \left(1 \pm \overline{\sigma}_{\mathrm{err},\nu}\right) \nu_\mathrm{b} \end{bmatrix}$$

In the set of experiments being simulated here, and in order not to overload this section and repeat experiments, we chose a relative error that is shared by each parameter (Poisson's ratio and Young's modulus). Likewise, the ϕ ratio is assumed to be the same for both parameters.

$$\overline{\sigma}_{\mathrm{err},\mu} = \left|\overline{\sigma}_{\mathrm{err},E}\right| = \left|\overline{\sigma}_{\mathrm{err},\nu}\right|, \quad \text{and} \quad \phi = \phi_E = \phi_\nu$$

In practice, we focus on the HF cost functional :

$$J^{\mathrm{hf}}\left(\mu = [E, \nu]^\top\right) = \frac{1}{\phi^2}\left(\frac{E - E_\mathrm{b}}{\Lambda_E}\right)^2 + \frac{1}{\phi^2}\left(\frac{\nu - \nu_\mathrm{b}}{\Lambda_\nu}\right)^2 + \left\|\mathbf{y} - H\left(\mathbf{u}_\mu\right)\right\|_2^2$$

With this formalism, it is then also possible to design the matrix A (cf. Equation (5.15)) used to normalize the constraint in the case of a classical trust-region approach:

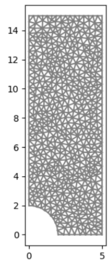
$$\mathbf{A} = \begin{bmatrix} 1/\Lambda_E^2 & 0 \\ 0 & 1/\Lambda_\nu^2 \end{bmatrix}$$

We examine the simplest case of an observation operator on the displacement field. Therefore, we adopt the observation operator equal to the identity. Since this operator is linear, it is equal to its tangent operator. In short, we get :

$$H\left(\mathbf{u}\right) = \mathbf{u}, \quad \text{then} \quad \left\{ \begin{array}{lcl} H & = & I_d \\ \mathbf{H} & = & \frac{\partial H}{\partial \mathbf{u}} = I_d \end{array} \right.$$

5.5.3 FE discretization and geometry

We consider a holed plate loaded with a tensile force in two dimensions. The geometry is a rectangle of size $(2L_x) \times (2L_y)$, perforated at its center by a circle of radius r. For symmetry considerations, we shall consider a quarter of the structure and apply the corresponding boundary conditions. To achieve this, we assume that vertical displacements are blocked on the bottom face, and that horizontal displacements are blocked on the left face. A visualization of the mesh is given in Figure 5.5a, while the dimensions of the geometry and informations about the mesh are provided in the Table given in Figure 5.5b.

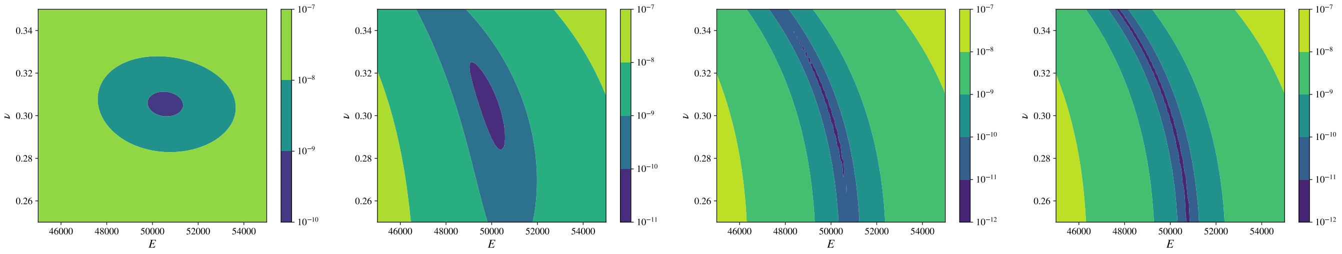


(a) HF mesh used for the holed plate test case.

L_x	L_y	r	N_e	$\mathcal{N}$
5	15	2	1096	4594

(b) Summary of the information (mesh and geometry).

Figure 5.5: Information about the geometry and the mesh for a holed plate.

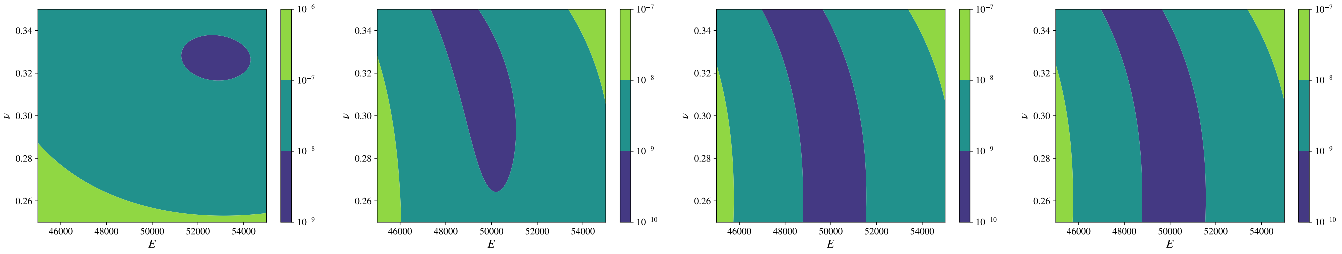


(a) $\phi = 1$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(b) $\phi = 10$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(c) $\phi = 100$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(d) $\phi = 10^3$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.



(e) $\phi = 1$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

(f) $\phi = 10$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

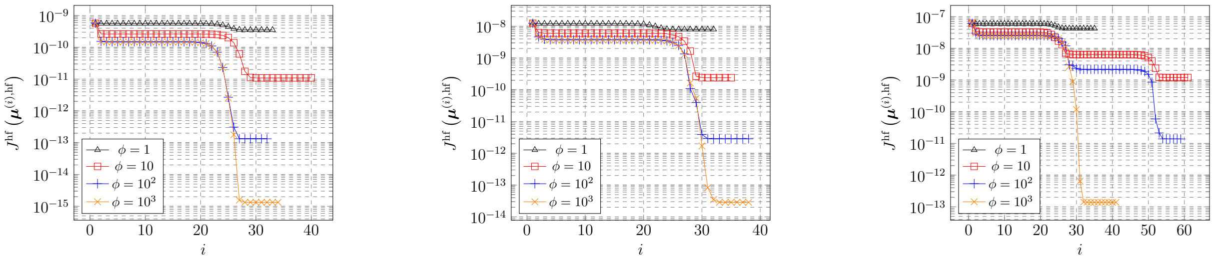
(g) $\phi = 100$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

(h) $\phi = 10^3$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

Figure 5.6: Plot of level lines (with interpolation) related to the cost function for the HF simulation (J^{hf}) for different sets of hyper-parameters (weights between the two contributions and initial errors between the background and real parameter).

Visualisation of the data-assimilation problem for the HF formulation

The definition of an optimization problem set previously (see Section 5.5.2) provides two hyper-parameters ($\phi, \overline{\sigma}_{\mathrm{err},\mu}$) to uniquely define a DA problem. As ϕ tends towards 0, we tend towards an increasingly convex problem, based solely on the regularization term. When ϕ tends towards ∞, the problem tends towards the true inverse problem. The second hyperparameter $\overline{\sigma}_{\mathrm{err},\mu}$ defines the gap between the background and the true parameter. When the two values are close, the two terms of the functional do not compete. In the opposite case, the sum of the two functionals may provide a total non-convex functional.



(a) $\overline{\sigma}_{\text{err},\mu} = 2\%$

(b) $\overline{\sigma}_{\text{err},\mu} = 10\%$

(c) $\overline{\sigma}_{\text{err},\mu} = 25\%$

Figure 5.7: Evolution of the cost function for HF simulation for different sets of hyper-parameters (weights between the two contributions and starting errors between the background and real parameter) using 'L-BFGS-B' method to solve the sub-problem.

Figure 5.6 features examples of cost functions obtained for different hyper-parameter pairs. In addition, Figure 5.7 displays the convergence curves towards the global minima obtained for a solution using L-BFGS-B [BLNZ95][ZBLN97] with an adjoint method for an HF case. We have examples where the BFGS algorithm converges to a local minimum. We notice that the value of the minimum varies as a function of the ϕ hyper-parameter. This is to be expected, since when ϕ tends to infinity, we tend towards an inverse problem with no regularisation term. For a more detailed visualisation of the algorithm's outputs, we provide in Appendix E.1 the results obtained (cf. Table E.1) for our DA algorithms on the stated HF problem (recalibration using the complete field in linear elasticity).

5.5.4 Adjoint approach for a trust-region approach with ROM and without trust-regions

(a) Young's modulus (E).

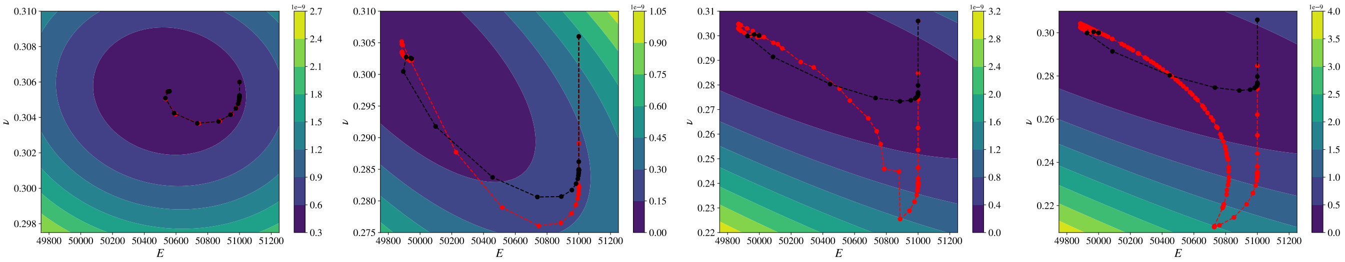
(b) Poisson coefficient (ν).

Figure 5.8: Comparison between the parameter values obtained for a reduced case with infinite radius and the parameters obtained by solving the HF problem for the case $\overline{\sigma}_{\text{err},\mu} = 2\%$ with enrichment considering all available information ($\varepsilon_{\text{POD},u} = 0$).

First, we perform tests without using trust-regions. In other words, we assume a trust-region with infinite radius at every iteration ($\Delta_k = \infty$). Thus, the point is to consider ROMs, enriched iteratively, each time solving the optimization problem over the entire parametric domain ($\mathcal{R}_k = \mathcal{P}$).

This validation stage of the methodology has two key interests: to check that the various components of the computational code are working properly (purely computational), and to highlight the interest of the trust-region radius in the algorithm. The numerical results reported herein (and in the rest of the Chapter) involve the construction of a reduced basis, enriched iteratively, with zero POD tolerance ($\varepsilon_{\text{POD},u} = 0$). In other words, we keep all modes available throughout the iterations. The reason for this choice is that we deal with a static problem (with a single snapshot per parameter) and the reduction on the parameter space does not require excessively large bases. Some intuitions about the behaviour of the algorithm with a non-zero tolerance are provided in the Appendix E.1 (see Section E.1.2). Even with a non-zero tolerance, the algorithm converges sufficiently close to the solution, but an additional error is introduced. Further work is being carried out to explore the consequences of using more restrictive tolerance in quasi-static cases, and the building of an adaptive basis if needed.

We carried out these test cases and compared the numerical results with the solutions obtained using the HF calculation code. Figure 5.8 depicts the relative errors between the iterates obtained during the optimization process with ROM and the final solution obtained in the HF case for the particular case of one value of the $\overline{\sigma}_{\text{err},\mu}$ parameter. We opted to limit the number of illustrations so as not to overload this section. We may observe that we converge towards the same solution as for the HF case, which indicates that the construction of an iteratively enriched ROM is effective for the cases studied here. On these curves, the number of iterations refers to the total number of optimization iterations (not trust-region iterations). At certain points on the curves, particularly at the start of iterations and for high ϕ values (problem not very regularized), we notice that the relative error can increase.



(a) $\phi = 1$ and $\overline{\sigma}_{\text{err},\mu} = 2\%$.

(b) $\phi = 10$ and $\overline{\sigma}_{\text{err},\mu} = 2\%$.

(c) $\phi = 100$ and $\overline{\sigma}_{\text{err},\mu} = 2\%$.

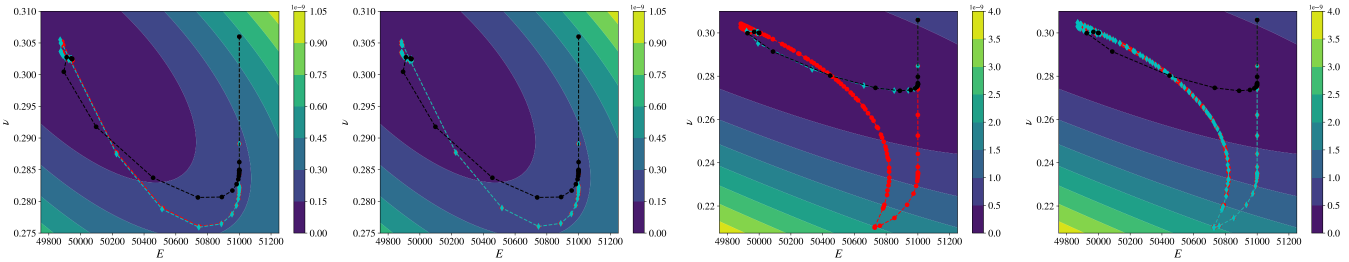
(d) $\phi = 10^3$ and $\overline{\sigma}_{\text{err},\mu} = 2\%$

Figure 5.9: Evolution of trajectories obtained by the optimization process for a case using a successively reduced model with an infinite trust-region radius. The curve in **black** corresponds to the HF case, while the curve in **red** corresponds to the iterates for the reduced case with enrichment considering all available information ($\varepsilon_{\text{POD},u} = 0$).

Such behavior is due to the fact that the ROM is of poor quality over the entire parametric domain, and does not reflect the correct variations in the cost functional. This is illustrated in Figure 5.9. In the latter, we plot the trajectories of the search of the optimum in the parametric space. The colormap in the background reflects the HF cost function used (J^{hf}). We observe that for large ϕ values in particular, there's a real gap between the two values. What's more, for the first ROM used (first iteration of TR), the trajectory evolves towards a region where the HF cost function is higher than at the start. Nevertheless, ROM enrichment corrects this deviation, and moves the trajectory towards the true solution. Therefore, the effect of deviation on relative errors can be seen visually in Figure 5.9.

5.5.5 Adjoint approach for a trust-region approach with ROM and with trust-regions

We now turn to the methodology as a whole, i.e. the approach with trust-region and ROMs. In view of the details given above, we have two methodologies to work with: on the one hand, the classical approach (`'classical'`), and on the other, the error-aware approach (`'errind'`), based on an error indicator.



(a) $\phi = 10$ and $\Delta_0 = 10^{-1}$.

(b) $\phi = 10$ and $\Delta_0 = 1$.

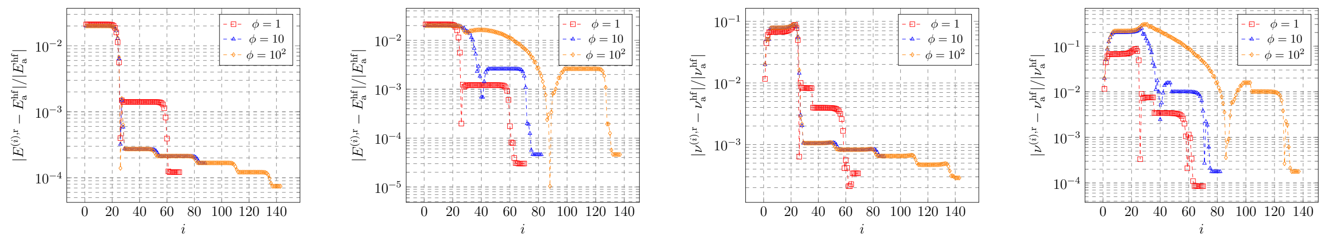
(c) $\phi = 10^3$ and $\Delta_0 = 10^{-1}$.

(d) $\phi = 10^3$ and $\Delta_0 = 1$.

Figure 5.10: Evolution of trajectories obtained by the optimization process for $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$ for different algorithms. The curve in **black** corresponds to the HF case. The curve in red corresponds to the iterates for the reduced case with enrichment considering all available information ($\varepsilon_{\mathrm{POD},u} = 0$) for $\Delta_k = \infty$. The curve in **cyan** corresponds to the iterates for the reduced case with enrichment considering all available information ($\varepsilon_{\mathrm{POD},u} = 0$) for a `'classical'` trust-region algorithm.

First, we consider the case of a `'classical'` trust-region approach, that is to say using a Euclidean distance (with adimensionalization). Figure 5.10 illustrates the interest of the trust-region on the trajectory. In particular, we can see the difference between an initial radius $\Delta_0 = 1$ and a initial radius $\Delta_0 = 10^{-1}$ for a value of $\phi = 10^3$ (close to an unregularized inverse problem). Restricting the resolution domain prevents the trajectory from deviating too much from the trajectory obtained for an HF case. The analysis we make of this observation is as follows: the fact of having a small initial radius implies that we prevent the ROM from heading into regions where the ROM loses validity.

In terms of relatives errors, Figure 5.11 provides the decreases in errors between iterates for the ROM and in the HF case. We notice that the values tend towards zero and thus that the parameter values obtained are approximately the same as in the HF case (maximum error up to the percent). These two remarks illustrate the interest of the approach with ROM and trust-region: first, we maintain a good quality of approximation, and second, the addition of the trust-region allows the trajectory no to deviate in areas where the ROM has no validity.



(a) Error on E for $\Delta_0 = 10^{-1}$.

(b) Error on E for $\Delta_0 = 1$.

(c) Error on ν for $\Delta_0 = 10^{-1}$.

(d) Error on ν for $\Delta_0 = 1$.

Figure 5.11: Relative errors between the points obtained during the reduced DA procedure $(E^{(i),\mathrm{r}}, \nu^{(i),\mathrm{r}})$ and the analysis obtained for the HF DA $(E_\mathrm{a}, \nu_\mathrm{a})$ and by taking all the available information ($\varepsilon_{\mathrm{POD},u} = 0$).

5.5.6 Numerical examples for alternatives resolution algorithms for the sub-problem

We intend to enhance the robustness of the approach by using other approaches for the sub-problem solvers. Indeed, it may be interesting to provide an implementation for cases where extracting the matrices for an adjoint calculation may be an issue. This situation may appear when applying the methodology to the industrial-grade FE code `code_aster`. First, we use a first-order centered finite-difference gradient calculation. Namely, each component of the gradient is computed according to the classical finite centered difference rule:

$$\forall \mu \in \mathbb{R}^p, \quad (\nabla J(\mu))_i = \frac{\partial J}{\partial \mu_i}(\mu) \approx \frac{J(\mu + \delta\mu_i \mathbf{e}_i) - J(\mu - \delta\mu_i \mathbf{e}_i)}{2\delta\mu_i}$$

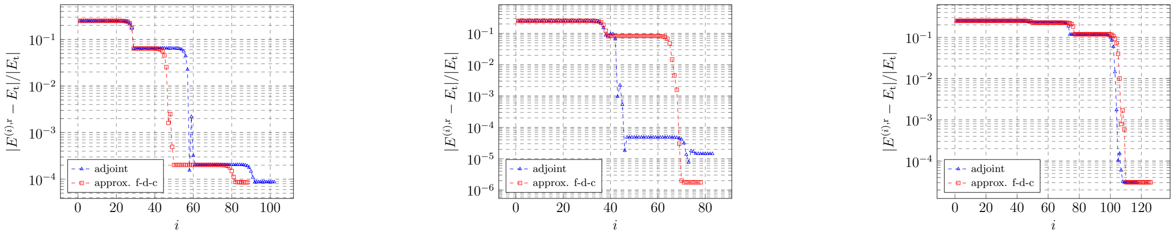
where $\mathbf{e}_i$ is the i-th euclidian vector basis and $\delta\mu_i$ is sufficiently small. The approach is designed not only to test the quality of gradient calculations using adjoint methods, but also to exploit an approach requiring gradients for cases where explicit adjoint calculations are not feasible.

Use of an approximate gradient for L-BFGS-B

Figure 5.12 depicts the convergence of the solution by showing the decrease in relative errors over the iterations. There is a small difference in values of relative errors throughout the iterations, depending on the set of hyper-parameters, but the resulting final value is similar.

Use of gradient-free approach for sub-problem resolution

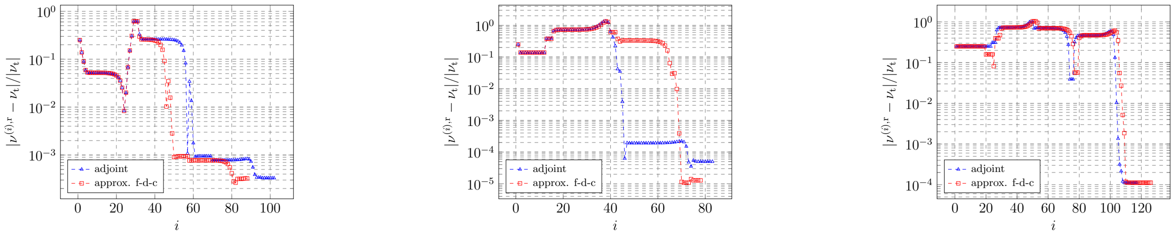
In addition, we investigate several gradient-free resolution methods. We rely on Python's `scipy` package, and in particular report results for two algorithms: Nelder-Mead and Powell. The Nelder-Mead simplex algorithm [NM65] stands as a popular approach for optimizing multidimensional unconstrained functions. Categorized within the broader class of direct search algorithms, these methods rely on evaluating the function at a set of sample points and then employ this information to guide further exploration. The Nelder-Mead minimization technique is predicated on assessing the function values at $(n + 1)$ vertices of a general simplex. The simplex is subsequently modified through operations like reflection, expansion, and contraction, aimed at replacing the vertex with the highest objective function value with a point that yields a lower value. In contrast, the Powell



(a) Relative error on E for $\Delta_0 = 1.0$.

(b) Relative error on E for $\Delta_0 = 10^{-1}$.

(c) Relative error on E for $\Delta_0 = 10^{-2}$.



(d) Relative error on ν for $\Delta_0 = 1.0$.

(e) Relative error on ν for $\Delta_0 = 10^{-1}$.

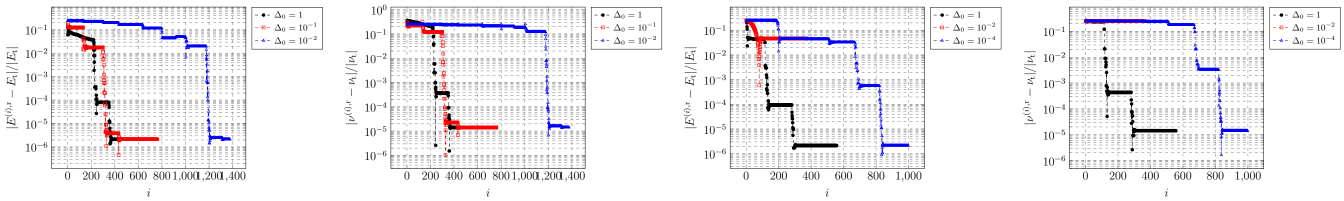
(f) Relative error on ν for $\Delta_0 = 10^{-2}$.

Figure 5.12: Relative errors between iterates and actual values of physical parameters for a DA-TR-ROM problem for a BFGS approach with exact gradient and approximate gradient (`'classical'` approach) for $\overline{\sigma}_{\mathrm{err},\mu} = 25\%$.

method is a modification of Powell's conjugate direction method [TFPV92][Pow64], a numerical optimization technique used to minimize a function. In Powell's method, the objective is to find the minimum of a function, and it is distinct from traditional gradient-based methods because it doesn't require the function to be differentiable, and it doesn't involve computing derivatives. Instead, it performs sequential one-dimensional minimizations along a set of search vectors, which are updated at each iteration during the main minimization loop (using golden-section search for instance).

Comparison of different resolution algorithms

In conclusion, we propose a comparison of the results for all the approaches used. Figure 5.14 displays the decreasing values of the cost functions for the two approaches (classical and error-aware). We notice that the values of the cost functions at convergence are of a similar order of magnitude. Figure 5.15 illustrates the decrease in relative values (comparison between iterated and actual values). All algorithms converge to the same value for all approaches, demonstrating the robustness of the implementation.

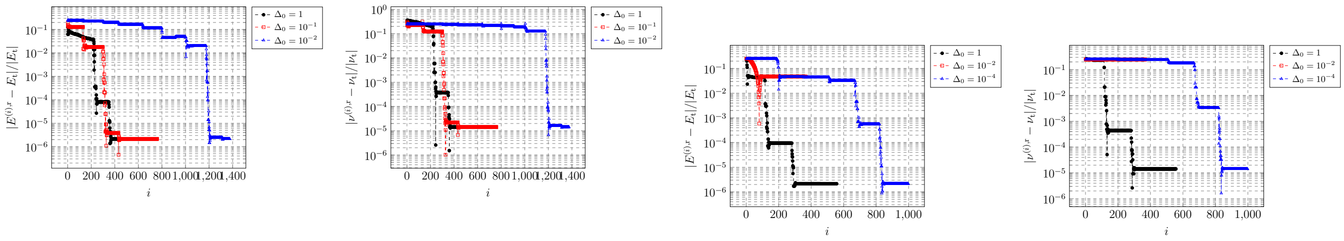


(a) Relative error on E for Nelder-Mead solver (`'classical'` approach).

(b) Relative error on ν for Nelder-Mead solver (`'classical'` approach).

(c) Relative error on E for Nelder-Mead solver (`'errind'` approach).

(d) Relative error on ν for Nelder-Mead solver (`'errind'` approach).



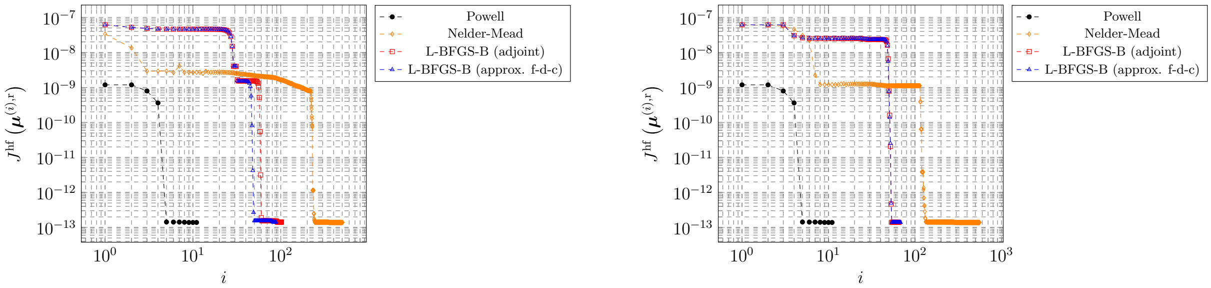
(e) Relative error on E for Powell solver (`'classical'` approach).

(f) Relative error on ν for Powell solver (`'classical'` approach).

(g) Relative error on E for Powell solver (`'errind'` approach).

(h) Relative error on ν for Powell solver (`'errind'` approach).

Figure 5.13: Relative errors between iterates and actual values of physical parameters for a DA-TR-ROM problem for different gradient-free solving algorithms for different initial radii Δ_0 for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.



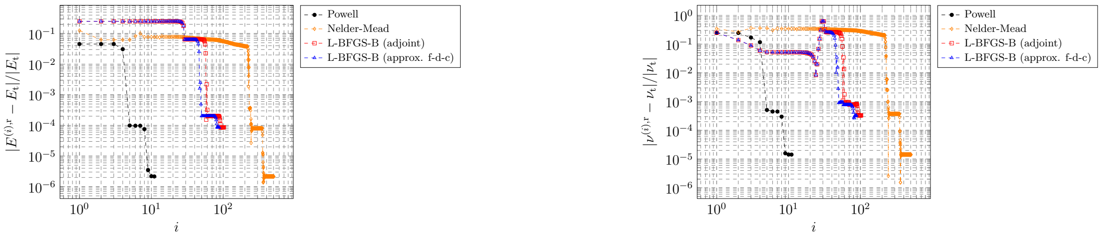
(a) `'classical'` approach.

(b) `'errind'` approach.

Figure 5.14: Evolution of the HF cost function for a DA-TR-ROM problem for different solution algorithms (with exact gradients, with approximate gradients and without gradients) for an initial radius of $\Delta_0 = 1.0$ for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.

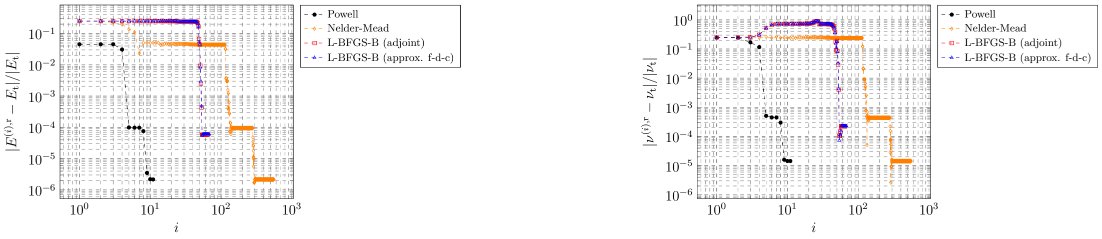
5.6 Numerical analysis of a data-assimilation problem for a hyperelastic material

In this section, we present numerical results an hyperelastic material. The objective of this study is to extend the results obtained in the previous section to a nonlinear case, while remaining in



(a) Relative error on E for `'classical'` approach.

(b) Relative error on ν for `'classical'` approach.



(c) Relative error on E for `'errind'` approach.

(d) Relative error on ν for `'errind'` approach.

Figure 5.15: Relative errors between iterates and actual values of physical parameters for a DA-TR-ROM problem for different solving algorithms (with exact gradients, with approximate gradients and without gradients) for an initial radius of $\Delta_0 = 1.0$ for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.

a static case within the framework of elasticity. The formalism of hyper-elasticity checks these assumptions. First, we present the general formalism associated with a neo-Hookean hyper-elastic model. Second, we present the numerical case studied (geometry and mesh), before presenting numerical results.

5.6.1 Formulation of the physical problem

We provide hereine the HF formulation of an hyperelastic material. Such formalism is used in order to model highly deformable structures, and is founded on the definition of a energy which is defined as the quantity of elastic energy that the material stores as a function of the stretching imposed. Indeed, an hyperelastic problem can be expressed as a minimization problem. We aim at finding a $u_\mu \in \mathcal{X}$ that minimizes a given total potential energy $\Pi(u, \mu)$:

$$u_\mu = \arg\min_{u \in \mathcal{X}} \Pi(u, \mu), \text{ with } \quad \Pi(u,\mu) = \underbrace{\int_\Omega \psi(u,\mu)\ dx}_{\Pi^{\text{int}}} - \underbrace{\left(\int_\Omega f_v u\ dx + \int_{\Gamma_n} f_s u\ ds\right)}_{\Pi^{\text{ext}}}$$

where the ψ is the elastic energy density, f_v is a body force and f_s is a traction force. At the minimum point (u_μ) of the potential, its directional derivative with respect to change in the displacement field u_μ is equal to zero $\forall v \in \mathcal{X}^{\text{hf}}$:

$$\mathcal{G}(u_\mu, v, \mu) = D_v \Pi = \frac{d\Pi(u + \varepsilon v)}{d\varepsilon}\Big|_{\varepsilon=0} = 0, \tag{5.26}$$

In order to define the stored elastic energy density, we introduce the deformation gradient $F(u)$ and the Cauchy-Green tensor $C(u)$ for a given displacement u. Those tensors are needed in order to introduce invariants such as $I_c(u)$ and $J(u)$ (scalars), defined as follows:

$$\left\{\begin{array}{lcl} F(u) & = & I + \nabla u \\ C(u) & = & \left[F(u)\right]^{\top} F(u) \end{array}\right. \quad \text{and} \quad \left\{\begin{array}{lcl} J(u) & = & \det\left(F(u)\right) \\ I_c(u) & = & \operatorname{tr}\left(C(u)\right) \end{array}\right.$$

The choice of the hyper-elasticity model corresponds to a choice of the expression of the stored energy. The latter is expressed thanks to invariants. In our case, we consider a neo-Hookean type modeling, whose total potential energy is of the form:

$$\psi(u,\mu) = \frac{G}{2}\left(I_c(u) - 3\right) - G\ln\left(J(u)\right) + \frac{\lambda}{2}\ln\left(J(u)\right)^2 \tag{5.27}$$

When seeking to calculate the sensitivities of the residual, we decompose the variational form in order to have terms depending only on the field of displacement and others depending only on the parameter. We begin by the decomposition of the stored elastic energy:

$$\psi(u,\mu) = \frac{G}{2}\underbrace{\left(I_c - 3\right)}_{=\psi_1} - G\underbrace{\ln(J)}_{=\psi_2} + \frac{\lambda}{2}\underbrace{\ln(J)^2}_{=\psi_3} \tag{5.28}$$

This induces a decomposition of the total potential energy in three seperate terms:

$$\Pi^{\text{int}}(u,\mu) = \frac{G}{2}\underbrace{\int_{\Omega}\psi_1\, dx}_{\Pi_1^{\text{int}}} - G\underbrace{\int_{\Omega}\psi_2\, dx}_{\Pi_2^{\text{int}}} + \frac{\lambda}{2}\underbrace{\int_{\Omega}\psi_2\, dx}_{=\Pi_3^{\text{int}}} \tag{5.29}$$

The discrete residual can be thus decomposed thanks to three vectors $\mathbf{f}_i^{\text{int}}$ which are expressed as nonlinear combination of $\mathbf{u}$:

$$\mathbf{R}^{\text{hf}}(\mathbf{u},\mu) = \frac{G}{2}\mathbf{f}_1^{\text{int}}(\mathbf{u}) - G\mathbf{f}_2^{\text{int}}(\mathbf{u}) + \frac{\lambda}{2}\mathbf{f}_3^{\text{int}}(\mathbf{u}) - \mathbf{f}^{\text{ext}} \tag{5.30}$$

We reproduce the previous procedure by expressing this discrete residue using the components of the μ vector:

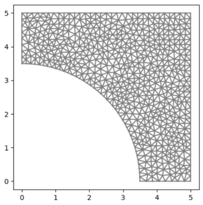
$$\begin{aligned} \mathbf{R}^{\text{hf}}(\mathbf{u},\mu) &= \frac{G}{2}\mathbf{f}_1^{\text{int}}(\mathbf{u}) - G\mathbf{f}_2^{\text{int}}(\mathbf{u}) + \frac{\lambda}{2}\mathbf{f}_3^{\text{int}}(\mathbf{u}) - \mathbf{f}^{\text{ext}} \\ &= G\left(\frac{1}{2}\mathbf{f}_1^{\text{int}}(\mathbf{u}) - \mathbf{f}_2^{\text{int}}(\mathbf{u})\right) + \frac{\lambda}{2}\mathbf{f}_3^{\text{int}}(\mathbf{u}) - \mathbf{f}^{\text{ext}} \\ &= \frac{E}{2(1+\nu)}\left(\frac{1}{2}\mathbf{f}_1^{\text{int}}(\mathbf{u}) - \mathbf{f}_2^{\text{int}}(\mathbf{u})\right) + \frac{E\nu}{2(1+\nu)(1-2\nu)}\mathbf{f}_3^{\text{int}}(\mathbf{u}) - \mathbf{f}^{\text{ext}} \end{aligned}$$

As a result, we dispose of a parametric affine decomposition of the discrete residual as in the case of linear elasticity. In this case, however, the internal force terms are nonlinear terms. Consequently, the gain in terms of CPU cost is not fully effective unless hyper-reduction is used. Nevertheless, we chose a nonlinear example in order to illustrate the good convergence of the algorithm in the case of introducing a nonlinearity. Therefore, for this numerical example, we decide not to use a hyper-reduction process.

5.6.2 Two-dimensional hyperelastic test case

Holed-squared under volumic loading

We consider a two-dimensional square subjected to a volumetric force. Figure 5.16 represents the mesh used for the HF calculations, and the set of parameters is given in Table 5.21. We use a vertical body force loading.



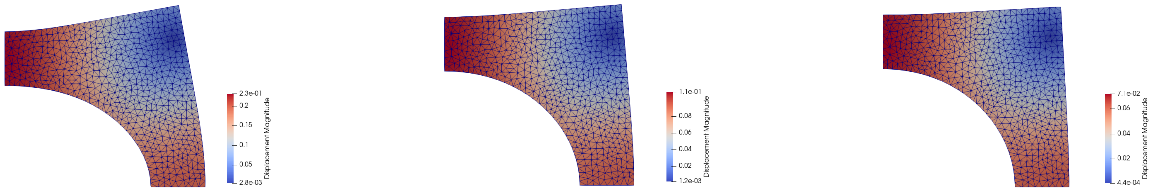
L_x	L_y	r	N_{e}	$\mathcal{N}$
5	5	3.5	1203	5060

(a) HF mesh used for the hyperelastic squared plate test case.

(b) Summary of the information (mesh and geometry).

Figure 5.16: Information about the geometry and the mesh for a holed square.

The type of geometry is quite similar to the case of linear elasticity, where we take into account a symmetrical geometry. Symmetrical boundary conditions entail that vertical displacements are restrained at the bottom, while horizontal displacements are restrained at the left. The rest of the edges are left free. Figure 5.17 represents some HF snapshots (with a multiplicative factor for visualization) to help observe the displacement field studied.



(a) $\mathbf{u}^{\mathrm{hf}}$ snapshot for $\mu = (E,\nu) = (25\cdot 10^3, 0.3)$.

(b) $\mathbf{u}^{\mathrm{hf}}$ snapshot for $\mu = (E,\nu) = (50\cdot 10^3, 0.3)$.

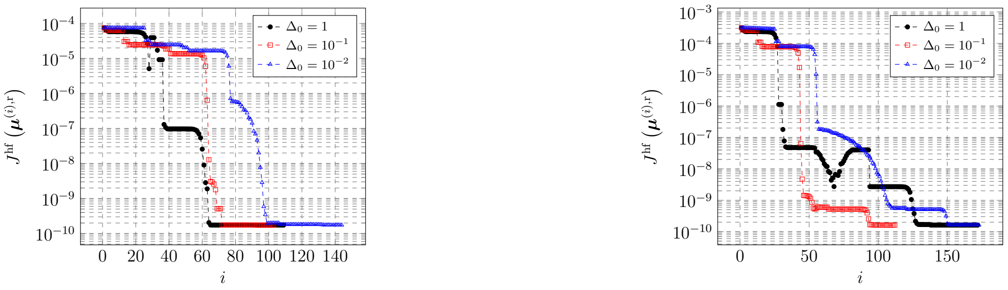
(c) $\mathbf{u}^{\mathrm{hf}}$ snapshot for $\mu = (E,\nu) = (75\cdot 10^3, 0.3)$.

Figure 5.17: HF snapshots obtained for different sets of parameters for the hyperelastic problem (×3 for visualisation).

In the present context, we shall consider two observation operators. First, we retain the simplest operator available and used previously, i.e. one that relies on the entire field (identity). Second, we look at an operator restricted to two points: the displacement at $(0, r)$ and $(r, 0)$, i.e. at the intersection between the hole and the vertical and horizontal axes. Hence, these examples provide an indication of both the extension of the methodology to a nonlinear case, and the use of a local observation operator, which stand for two ways of stiffening the optimization problem.

Numerical results on the holed-squared under volumic loading

In this section, we report some numerical results on this geometry and constitutive equation. The objective of this part is not to perform an extensive study of the algorithm's behavior on this case, but to extend the previous experiments to more complex situations to highlight the applicability of the methodology. In this respect, we restrict ourselves to the use of L-BFGS solver by adjoint method, as detailed in the previous sections.



(a) All field.

(b) Local observations.

Figure 5.18: Plot of the HF cost function estimated at iterates of the reduced problem, for different values of initial radii, comparing evolutions for a global or local observation operator for $(\overline{\sigma}_{\mathrm{err},\mu},\ \phi) = (25\%, 1000)$.

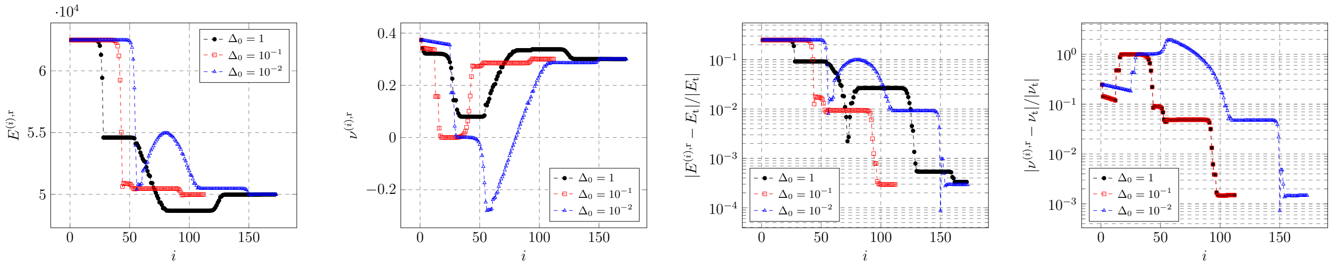
We achieve a decay of the cost functional and convergence towards an estimate of the parameters, whether for an operator over the whole field or for a local operator (cf. Figure 5.18). As shown in the linear case, the speed of convergence does depend on the initial radius provided (the method's hyperparameter), but the cost functional converges to a plateau of similar order of magnitude depending on the input radius. Furthermore, the plot of iterates and relative errors for a `'classical'` approach compared to the expected value clearly indicates convergence towards the desired value, and demonstrates the high quality of the method (cf. Figure 5.19).

In addition, these findings also apply to the case of an error-aware method (`'errind'`) featuring an error indicator, in which we converge on the desired solution (cf. Figure 5.20). However, we shall take the time to add a comment. We report here on results for two different initial radii, depending on the observation operators, since in this case the convergence of the algorithm is more sensitive to hyper-parameters, and *a priori* estimation of the right choice is not always obvious. Indeed, we rely on error indicators. All we have is a correlation between the approximation errors between the reduced model and HF and the error indicators. The order of magnitude of these two quantities are not necessarily the same, and the "slope" of the correlation is not known in advance. As a result, this method requires particular care when choosing hyper-parameters, although it does work and provides quality analyses.

5.6.3 Four-dimensional hyperelastic test case

Geometry and loadings

At last, we introduce a last elasticity case, also founded on hyper-elasticity. The objective of this test case is to remain with a nonlinear problem while increasing the dimension of the parameter

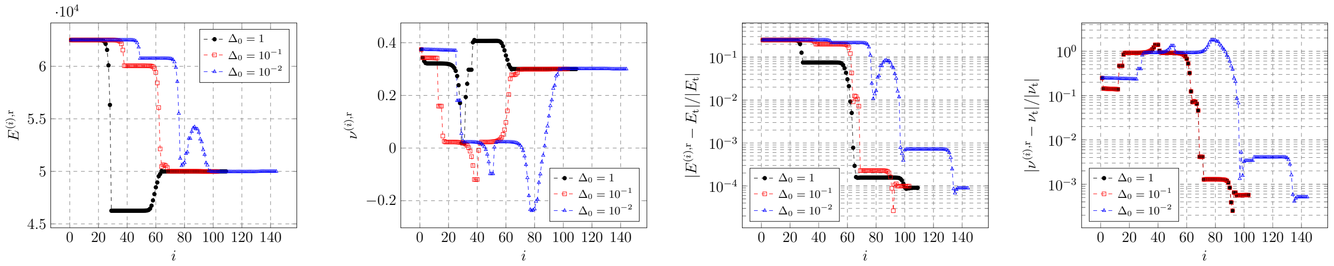


(a) E iterates local observation operator.

(b) ν iterates for local observation operator.

(c) Errors on E for local observation operator.

(d) Errors on ν for local observation operator.



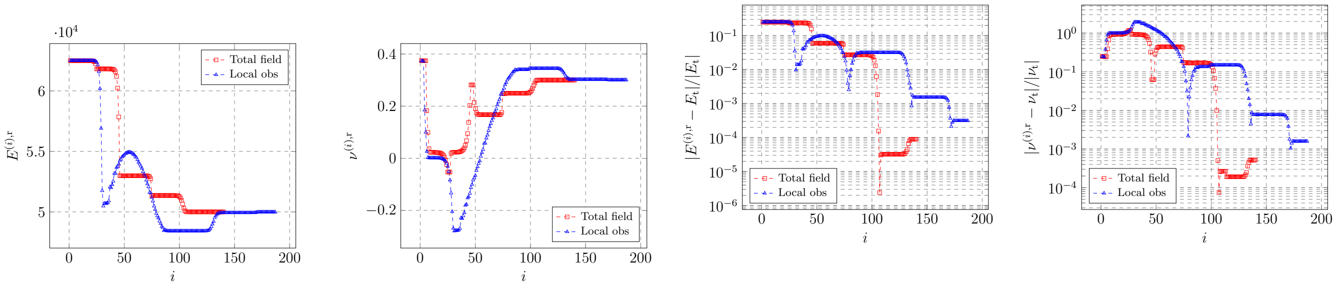
(e) E iterates total field observation operator.

(f) ν iterates for total field observation operator.

(g) Errors on E for total field observation operator.

(h) Errors on ν for total field observation operator.

Figure 5.19: Evolution of the parameters and the relative errors on the components of the parameter vector comparing to the true parameter for different values of initial radii for `'classical'` approach and both global and local observation operators for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.



(a) E iterates.

(b) ν iterates.

(c) Relative error on E.

(d) Relative error on ν.

Figure 5.20: Evolution of the parameters and the relative errors on the components of the parameter vector comparing to the true parameter for the `'errind'` approach and both global ($\Delta_0 = 1$) and local observation operators ($\Delta_0 = 10$) for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.

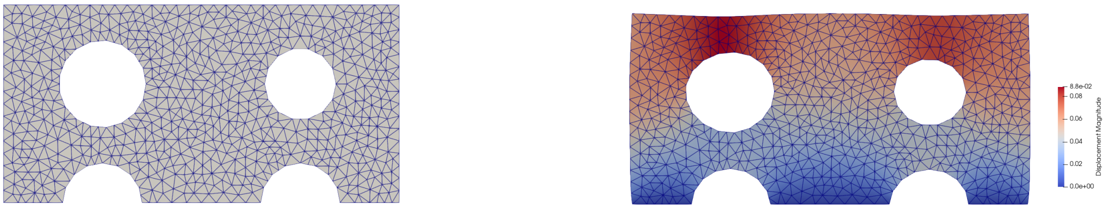
space. In order to achieve this goal, we consider a rectangular domain with several holes subjected to a volume force. Circular holes do not have a similar radius across all four holes. The two holes (half of a circle) at the bottom of the domain have radius r, while the hole in the middle of the domain on the left (resp. right) has a radius of size $r\phi_{\text{l}}^{\text{ratio}}$ (resp. $r\phi_{\text{r}}^{\text{ratio}}$). Therefore, we have a

L_x	L_y	r	$\phi_{\mathrm{r}}^{\mathrm{ratio}}$	$\phi_{\mathrm{l}}^{\mathrm{ratio}}$	N_{e}	$\mathcal{N}$
5	5	1	1.1	0.9	1252	5334

Figure 5.21: Summary of the information (mesh and geometry) for the hyperelastic rectangle test case.

geometric asymmetry. In addition to this geometric asymmetry, the material has an asymmetry in terms of physical parameters. The perforated rectangular block is separated vertically with two distinct hyper-elastic materials, one formed by a pair of material parameters (E_0, ν_0), and the other by a second pair (E_1, ν_1). Figure 5.22 displays the numerical values defining the geometric domain are provided. The visualisation of the mesh and an example of HF snapshot is provided on Figure 5.22. As a result, we artificially created a test case with a nonlinearity in terms of mechanical behavior, asymmetry and a 4-dimensional parameter space. In this case, the vector of parameters studied is defined as follows:

$$\mu = [E_0,\ \nu_0,\ E_1,\ \nu_1]^\top \in \mathbb{R}^4$$



(a) HF mesh used.

(b) Example of HF snapshot.

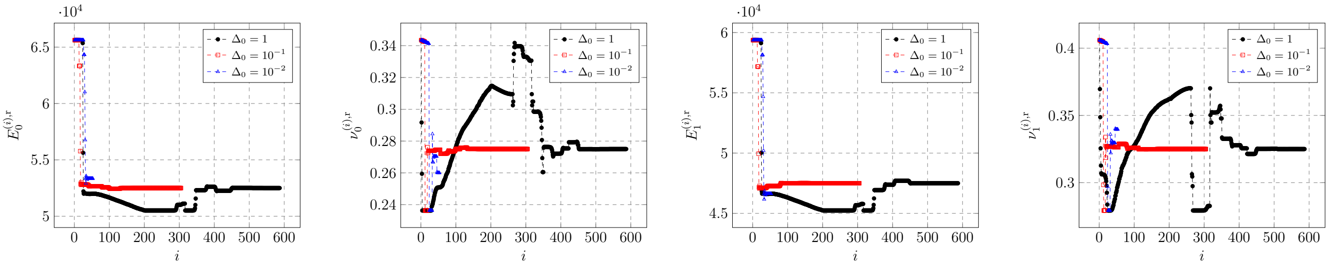
Figure 5.22: Visualisation of mesh and snapshots for the hyperelastic rectangle test case: HF mesh (cf. Figure 5.22a) and HF snapshot obtained for a given set of parameters (×3 for visualisation) for $(E_0, \nu_0, E_1, \nu_1) = (52.5e3, 0.275, 47.5e3, 0.325) \in \mathbb{R}^4$ (cf. Figure 5.22b)

Numerical validation for the four-dimensional parameter with a nonlinear behavior

We present numerical outcomes for a 4-dimensional parameter, emphasizing that our objective is not an exhaustive analysis of the test case but rather an extension of prior efforts to showcase the method's effectiveness in more complex scenarios. The heightened dimension of the current parameter could stiffen the optimization problem or introduce multiple local optima. Hence, for such scenarios, we have incorporated the use of bounds to tackle these potential complexities ($\mu_i \in [\mu_i^{\min}, \mu_i^{\max}]$). We refrain from elaborating further on this matter, as the employment of bounds for convergence facilitation is a well-established practice in optimization. In our approach, the selection of an alpha parameter allows us to delineate the resolution domain for each parameter:

$$\mu_i^{\max} = \mu_{\mathrm{b}} + \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}}, \quad \text{and} \quad \mu_i^{\min} = \mu_{\mathrm{b}} - \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}} \tag{5.31}$$

It is crucial to recognize that this approach is applicable in synthetic experiments; in real-world situations, such intervals would typically be determined based on engineers' expertise.

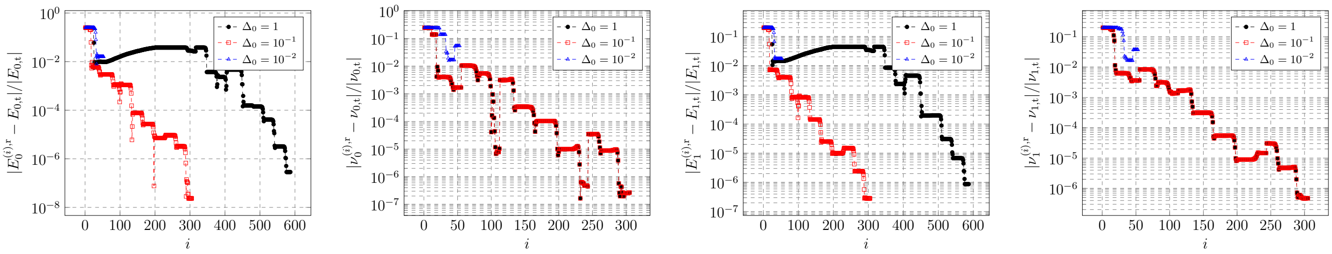


(a) E_0 iterates.

(b) ν_0 iterates.

(c) E_1 iterates.

(d) ν_1 iterates.



(e) Errors on E_0 iterates.

(f) Errors on ν_0 iterates.

(g) Errors on E_1 iterates.

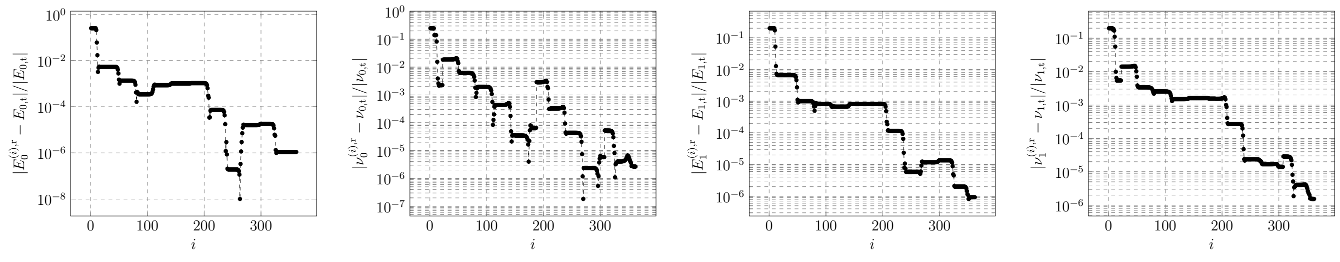
(h) Errors on ν_1 iterates.

Figure 5.23: Evolution of the parameters and the relative errors on the components of the parameter vector comparing to the true parameter for different values of initial radii for `'classical'` approach on the overall field for a four dimensional test case for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$.

Similar to previous cases, we attain a robust approximation quality for the optimization problem across different initial region values in the `'classical'` setting. In this context, we highlight that convergence quality remains consistent for all four studied parameters (cf. Figure 5.23). Furthermore, similar results are obtained when employing an error-aware approach (cf. Figure 5.24). We reiterate the significance of hyperparameter selection for this approach. The algorithm exhibits heightened sensitivity to this hyperparameter in the error-aware scenario (hence the limited presentation of results in this case). Nevertheless, a well-informed choice becomes feasible once the correlation between the indicator and approximation error is established.

5.7 Numerical analysis of data-assimilation using `code_aster`: application to an elasto-plastic material

In this section, we embark on an initial exploration of the trust-region methodology applied to the reduction approach developed within `code_aster`. The primary objective is to showcase the applicability of this method within industrial numerical tools, aligning with prior research endeavors. From an algorithmic standpoint, we seek to ascertain the compatibility of these algorithms with our code structure. In terms of problem complexity, this approach allows us to numerically explore two avenues: firstly, the complexification of behavior by incorporating quasi-static aspects and internal variables, and secondly, the consideration of the hyper-reduction process. To do so, we choose to illustrate the methodology's first step with the case of the elasto-plastic plate. This strategy aligns with the same philosophy used for model reduction, aiming for a gradual increase



(a) Errors on E_0 iterates.

(b) Errors on ν_0 iterates.

(c) Errors on E_1 iterates.

(d) Errors on ν_1 iterates.

Figure 5.24: Evolution of the parameters and the relative errors on the components of the parameter vector comparing to the true parameter for $\Delta_0 = 10$ for `'errind'` approach on the overall field for a four dimensional test case for $(\overline{\sigma}_{\mathrm{err},\mu},\ \phi) = (25\%, 1000)$.

in complexity in the cases studied. To maintain the original formulation, we restrict our analysis to the final state during the recalibration process, thus remaining within the framework of twin experiments. The ultimate goal is to determine the parameters used for data generation, given the knowledge of the final state.

5.7.1 HF calibration example for an elasto-plastic material

We start by defining the twin experiment, utilizing an elasto-plastic behavior law similar to the one outlined in the previous section of the chapter. The emphasis is on adjusting a vector $\mu \in \mathbb{R}^4$ containing two elasticity parameters and two hardening parameters. Thus, the background vector is defined as follows:

$$\mu_{\mathrm{b}} = \left[E_{\mathrm{b}}, \nu_{\mathrm{b}}, a_{\mathrm{pui,b}}, \sigma_{y,\mathrm{b}}\right]^{\top} \in \mathbb{R}^4$$

Initially, observe that the choice of the adimensionalization process is tailored to elasticity examples. The approach presented here proves more robust and is recommended for the expansion of our research. Practically, recalibration relies on using a multiplier $\lambda_\mu \in \mathbb{R}^4$ rather than concentrating on physical parameters. We choose to consider the background multiplier as the reference for multipliers, that is to say $\lambda_{\mathrm{b}} = [1, 1, 1, 1]^{\top}$. The key is to consider the inverse problem, where the regularisation term becomes $\|\lambda_\mu - \lambda_{\mathrm{b}}\|_2$, in order to seek the parameter multiplier λ_μ, defined as:

$$\mu = \begin{bmatrix} \lambda_E E_{\mathrm{b}} \\ \lambda_\nu \nu_{\mathrm{b}} \\ \lambda_{a_{\mathrm{pui}}} a_{\mathrm{pui,b}} \\ \lambda_{\sigma_y} \sigma_{y,\mathrm{b}} \end{bmatrix} \in \mathbb{R}^4. \tag{5.32}$$

For the tests presented here, we consider an approximate gradient-solving approach using finite differences. The goal is to validate the algorithm's compatibility with the problems at hand, and the non-optimality of relying on finite differences is not seen as a major concern.

The optimization calculations on a four-parameter model yield insightful results. There is commendable convergence in the elasticity parameters, with minimal changes observed in the plasticity parameters (cf. Figure 5.26). In the specific context considered, where only displacement field

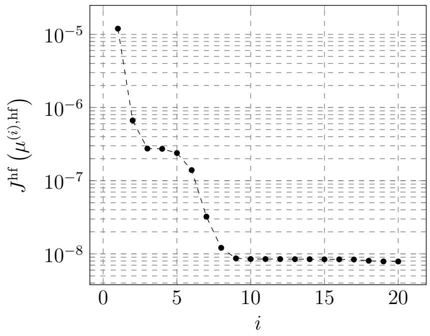
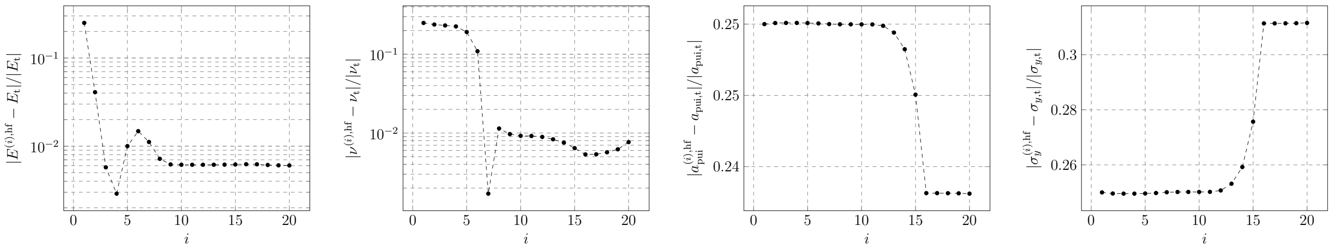


Figure 5.25: Evolution of the cost function on the components of the parameter vector comparing to the true parameter for a HF resolution on the overall field for a four dimensional elasto-plasticity test case for $(\overline{\sigma}_{\mathrm{err},\mu},\ \phi) = (25\%, 1000)$.

measurements are available and the system operates within a regime with limited plasticization, the logical outcome is a significant impact of elasticity parameter variations on the displacement field. This alignment with expectations leads to a noticeable reduction in the cost functional (cf. Figure 5.25). It's noteworthy that the absence of stress measurements, while physically inaccessible at present, limits the full recalibration of other parameters. This observation is not exclusively tied to ROM but is visualized in the HF case, suggesting potential opportunities for further calibration with stress measurements if available.



(a) Errors on E iterates.

(b) Errors on ν iterates.

(c) Errors on a_{pui} iterates.

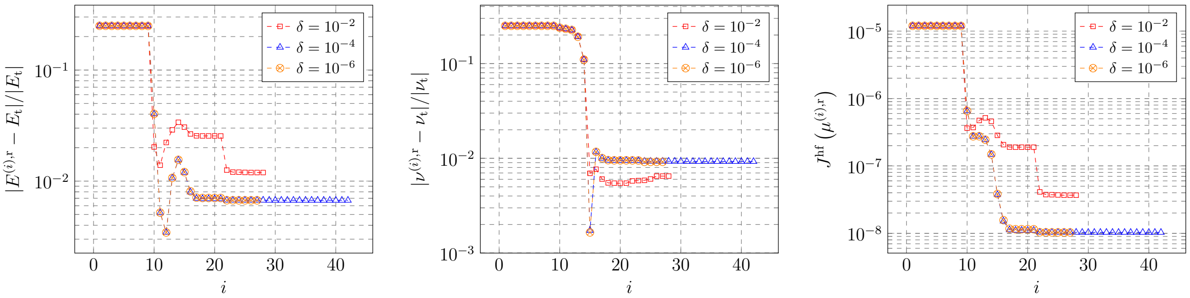
(d) Errors on σ_y iterates.

Figure 5.26: Evolution of the relative errors on the components of the parameter vector comparing to the true parameter for a HF resolution on the overall field for a four dimensional elasto-plasticity test case for $(\overline{\sigma}_{\mathrm{err},\mu},\ \phi) = (25\%, 1000)$.

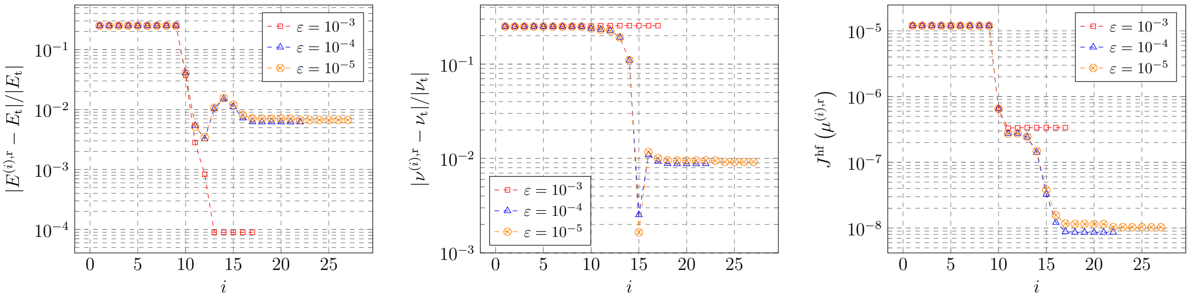
5.7.2 Preliminary tests for calibration using on-the-fly ROM adaptation with `code_aster`

The test case involves initial attempts at parameter calibration on a full field using on-the-fly hyper-reduced ROM adaptation with `code_aster`, focusing on cases with internal variables. The primary objective is not an exhaustive review but rather to gauge whether this algorithm could be a viable solution in the future. To assess this, we conducted a relatively straightforward case across the entire mechanical field, considering only the final state. Previous observations on the HF case highlight the significance of monitoring the evolution of the two elasticity parameters.

We specifically examine a full pod case with enrichment at each TR center μ_k.



(a) Errors on E iterates. (b) Errors on ν iterates. (c) Cost function decrease.



(d) Errors on E iterates. (e) Errors on ν iterates. (f) Cost function decrease.

Figure 5.27: Evolution of the relative errors on the parameter when using ECSW hyper-reduced ROMs with on-the-fly adaptation with a full POD (elasto-plasticity test case for $(\overline{\sigma}_{\text{err},\mu},\ \phi) = (25\%, 1000)$): Figures 5.27a-5.27b-5.27c are generated for a fixed POD tolerance and Figures 5.27d-5.27e-5.27f are generated for a fixed hyper-reduction tolerance.

Figure 5.27 illustrates two sets of variations: one with a fixed POD tolerance ($\varepsilon_{\text{POD},u}$) and the other with a fixed hyper-reduction tolerance (δ). These two hyperparameters, controlling the compression of the basis and meshing, are the key manipulable variables. As expected, for overly coarse tolerances ($\approx 10^{-2}$), the algorithm struggles to converge toward the correct solution for both tolerances. Notably, the POD tolerance exerts a more significant influence than the hyper-reduction tolerance in our case. However, for reasonable tolerances, the quality of the approximation achieved is comparable to that of a HF case. This is particularly intriguing as the number of iterations is close to that of a HF case, implying substantial computational gains provided that offline costs are optimized (hyper-reduction, reduced basis construction).

These initial indications suggest that the methodology is compatible with the hyper-reduced ROMs previously built. Nevertheless, a number of tests remain to be carried out in order to convince ourselves of its quality, notably by making the problem more complex. This would involve, for example, a highly plasticized case.

5.8 Conclusion

In this chapter, we presented a novel approach for constructing on-the-fly ROMs during iterations of an optimization problem. We have demonstrated that trust-region-based approaches, commonly applied in the literature for PDE-constrained optimization formulations in general, are well-suited for data assimilation problems, viewing them as optimization challenges. Numerically, we initially showcased the method's capability to solve elasticity problems, both linear and nonlinear, without the use of hyper-reduction. The approach yielded compelling results in terms of parameter convergence and iteration efficiency compared to HF cases. Furthermore, we established its adaptability to a variety of numerical solvers, with or without gradients, a crucial aspect for the generic application to industrial computational codes. This comprehensive study paved the way for initial tests on `code_aster`. Although not exhaustive, these tests indicate that the method is compatible with hyper-reduction strategies for more complex problems. Several avenues for future work are, however, worth exploring. Firstly, enhancing the calibration problem's complexity, whether in terms of physics (e.g., standard section) or observation localization, may require suitable adjustments in the on-the-fly construction. This ensures the method's capability to handle less well-posed inverse problems. Secondly, a more in-depth consideration of error indicators could prove valuable for leveraging information during the optimization process. Yet, this also emphasizes the need for robust indicators across a broader range of cases, particularly in multi-modeling scenarios.

Chapter 6

Acceleration of iterative procedures involving reduced-order models

6.1 Introduction

In the domain of applied mathematics and engineering, the justification for employing iterative processes is firmly grounded in theoretical underpinnings, showcasing their efficacy in practical implementations. These strategies include optimization algorithms (cf. Chapter 5) and greedy processes (cf. Chapter 2), e.g. for building ROMs. However, transitioning these theoretically developed approaches to industrial or real-world settings encounters significant challenges, primarily driven by computational costs. Indeed, when accouting for the computational costs within the online/offline paradigm (cf. Figure 1.5), one may not underestimate offline computations even if online computational gains are substantial. Indeed, it can become a critical consideration in addressing real-world engineering problems.

This chapter explores the intricate relationships between theoretical foundations, practical efficiency, and the obstacles hindering real-world applicability in iterative processes. Through examples drawn from our research, where the offline/online paradigm is applied, implementation and use within an industrial context may encounter a number of practical limitations. Notably, while greedy algorithms prove optimally or near-optimally efficient in numerous scenarios, their inherently sequential nature can lead to prohibitively high computational costs, especially when individual unit computations are resource-intensive. This is particularly the case when using industrial-grade code such as `code_aster`, with its complex three-dimensional models, and the costs associated with initializing data structures. Furthermore, even with the substantial acceleration achieved through the introduced hyper-reduction methodology for online computations, processes involving EQ procedure rely on optimization problems during the offline stage (cf. Equation (2.38)). The computational cost for solving such problems escalates with an increase in the number of rows (representing snapshots and modes: essentially the size of the approximation and training spaces) and columns (indicating mesh size).

Despite these challenges, even on-the-fly methods operating beyond the offline/online paradigm encounter similar issues. On-the-fly construction of reduced order bases for optimization problem, may lead to bases, whose size can expand throughout iterations, potentially compromising speedup gains. While the methodologies presented in this thesis offer undeniable advantages and pave the way for applications in real-world scenarios, their optimization is imperative for genuine applicability with substantial computational cost reductions. The objective of this chapter is to explore avenues for accelerating iterative processes involving ROMs, with the ultimate goal of easing their practical utilization. We delineate three primary research directions for accelerating computational problems. The ideas provided hereafter emerged from the study of greedy processes, but may also be applicable to optimization problems. First, we could explore the feasibility of employing a multi-fidelity approach to sample the parametric space. Specifically, we investigate meshes of varying resolutions (coarse or fine mesh resolution) to generate a sampling of the parametric space. Second, we introduce an acceleration strategy for optimization algorithms in the derivation of quadrature rules, specifically when employing an incremental data compression process. This approach is practical for greedy processes with incremental POD to significantly reduce offline costs. Moreover, it can serve as a suitable method for optimization problems involving enriched reduced bases built on a hierarchical foundation.

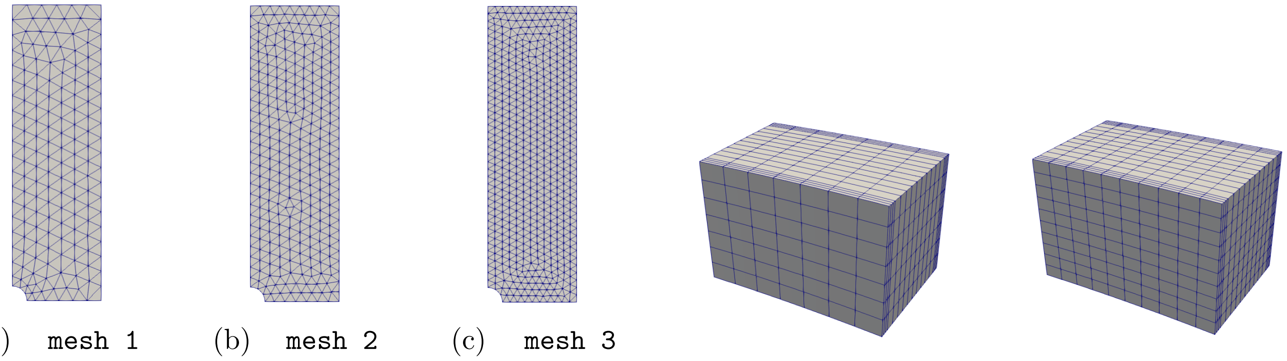
The theoretical and numerical aspects discussed in this section are based on content extracted from an online preprint, specifically drawing from the following publication:

- ◇ [AT24] Eki Agouzal and Tommaso Taddei. Accelerated construction of projection-based reduced-order models via incremental approaches. *arXiv preprint arXiv:2401.07108*, 2024.

6.2 Acceleration processes for greedy procedure

Building upon Veroy's pioneering work [VPRP03], numerous authors have applied greedy methods for parameter sampling. We may refer to Chapter 2 for more details on these approaches. Despite being widely used and extensively studied, these approaches may suffer from several limitations. On one hand, such algorithms are purely sequential and cannot leverage parallel computation to reduce CPU costs. Moreover, the offline computation cost at each iteration must be considered: this includes computing the reduced basis, constructing the empirical quadrature, or iterating over the training set to determine the least-approximated parameter. We present several approaches here to expedite the computation times of these different blocks, aiming to enhance the practical utility of these algorithms. Here, we introduce several approaches to expedite these processes. On one hand, we propose a multi-fidelity (specifically, bi-fidelity) approach to accelerate the sampling of the parametric space (see Section 6.2.1), and on the other hand, an incremental approach for computing the EQ rule (see Section 6.2.2).

To illustrate our methodologies, we revisit the two cases employed for constructing the reduced model, namely the elasto-plastic perforated plate and the standard section of a NCB. The geometry slightly differs for the perforated plate case, where the hole diameter is proportionally smaller concerning the plate size compared to the cases presented in Chapter 4. In order to present bi-fidelity approaches, we have at our disposal various meshes. For the elasto-plastic plate, we present three finely refined meshes uniformly covering the entire geometry (cf. Figures 6.1a, 6.1b, 6.1c). In the industrial case of the standard section of a NCB (cf. Figures 6.1d, 6.1e), we also applied a uniform refinement based on the mesh commonly used in practical industrial studies, where only a shape factor is increased. The selection of different mesh sizes is made to obtain finer meshes while striving to keep the evaluation cost manageable to conduct the tests presented in this thesis. The various meshes used and additional information are provided in Figure 6.1. The HF simulations of section 4.2 are performed using the FE software code aster and executed over a commodity Linux workstation (RAM 32 GB, Intel i7-9850H CPU 2.60 GHz x 12); on the other hand, the model order reduction procedure relies on an in-house `Python` code and is executed on a Windows workstation (RAM 16 GB, Intel i7-9750H CPU 2.60 GHz x 12).



(a) `mesh 1` (EP). (b) `mesh 2` (EP). (c) `mesh 3` (EP). (d) `mesh 1` (SS). (e) `mesh 2` (SS).

Figure 6.1: Presentation of the different three-dimensional meshes used for the calculations with `code_aster`. We have 3 meshes of plate with holes (elasto-plastic behavior) under tensile loading: : (a) coarse mesh (`mesh 1`: $N_\mathrm{e} = 1063$); (b) intermediate mesh (`mesh 2`: $N_\mathrm{e} = 5153$); (b) refined mesh (`mesh 3`: $N_\mathrm{e} = 10288$); and 2 meshes for a standard section of a power plant containment building: (a) coarse mesh (`mesh 1`: $N_\mathrm{e} = 784$); (b) refined mesh (`mesh 2`: $N_\mathrm{e} = 1600$).

6.2.1 Multi-fidelity approach for parameter sampling

Initially, we introduce a multi-fidelity approach, drawing inspiration from existing works wherein coarser simulations are employed to explore the parametric space [BTT24]. This methodology seeks to minimize the offline computational costs associated with HF simulations. More precisely, our approach is rather a bi-fidelity approach. Given two meshes, one finer and one coarser, parameter sampling using a greedy approach can be performed for both. Consequently, the algorithm provides an ordered list of parameters $\{\mu_j\}_j$ for exploration to construct a ROM. Application of this list to the finer mesh yields a ROM with the desired precision. Once the list of parameters has been obtained, it is worth assessing the quality of this sampling on the finest mesh. The bi-fidelity approach is therefore of interest when the sampling obtained on a coarse mesh provides a good approximation quality, or even a quality similar to that which would have been obtained by sampling directly with the fine mesh. More precisely, we compute the indices to explore across all possible meshes $\{\mathcal{T}_i^{\text{hf}}\}_{i\in\mathcal{I}_\bullet^{\text{comp}}}$ for $\bullet \in \{\text{ep}, \text{ss}\}$. The notation 'ep' refers to the numerical test cases associated with the elasto-plastic plate, while 'ss' is indicative of scenarios pertaining to the standard section. In order to acquire sets of indices, we first compute trajectories for all parameters in the training set Θ_{train} and across all available meshes. The, we can rely on a strong POD-Greedy relying on projection errors to sample the parametric space (see Algorithm 4).

Algorithm 4 Generation of an *a priori* sampling of the parameter space for a given HF mesh.

```
Require: a HF mesh $\mathcal{T}_i^{\text{hf}}$, and a training set $\Theta_{\text{train}}$
1: Generate HF data for the mesh $\mathcal{T}_i^{\text{hf}}$: $\mathbb{U}_{\mu,i}^{\text{hf}} = \left\{\mathbf{u}_{\mu,i}^{\text{hf},(k)}\right\}_{k=1}^{K}$, $\forall \mu \in \Theta_{\text{train}}$
2: Compute explored indexes through a POD strong-greedy process:        ▷ See Algorithm 5
       $\text{indexes}(i) = \text{POD-strong-greedy}\left(\left\{\mathbb{U}_{\mu,i}^{\text{hf}}\right\}_{\mu\in\Theta_{\text{train}}}\right)$
```

The detailed steps of a strong POD-Greedy approach grounded in projection errors are elucidated by Algorithm 5. This approach closely aligns with our prior work, particularly concerning the compression stages involving hierarchical bases (cf. Section 2.3.4). The significant divergence lies in the foundation on projection errors rather than approximation errors. The primary motivation behind this choice primarily stems from the desire for test case reproductibility in this section. In formulating the numerical experiment, we sought a framework where we could forego successive calls to `code_aster` for practicality reasons (and reproductibility if sharing data with a third party without installation). This is entirely achievable within this framework, as it only requires access to precomputed HF snapshots. Aside from this technical detail, the core of the method remains essentially identical.

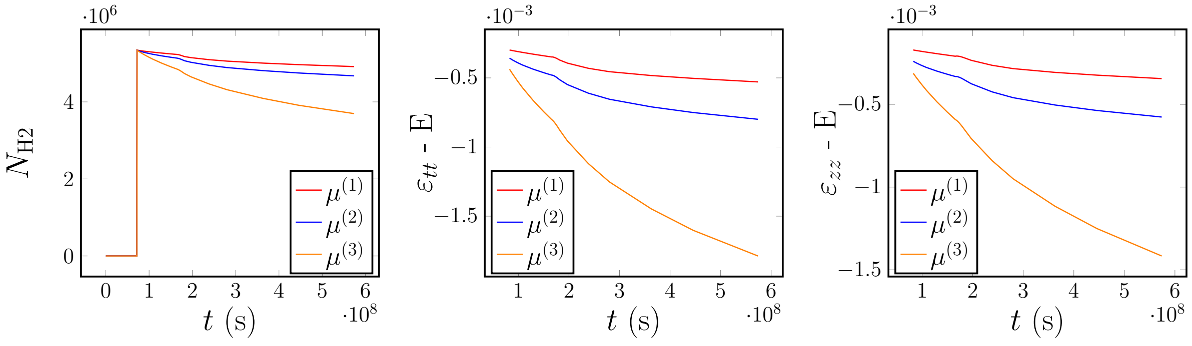
Our primary focus is directed towards our case of interest, specifically the standard section of a NCB. Before relying on the multi-fidelity strategy, we should have a comparative evaluation of solution quality across meshes of varying granularity. As elucidated in Chapter 4, these meshes are predominantly utilized for extracting quantities of interest (QoIs), notably normal forces within the bars and deformations at the sensor locations. Thus, we must have at our disposal a metric that compares these magnitudes between our finest mesh, considered as a reference, and the coarser mesh. Within this section, we study the solution behavior with respect to two parameters: the desiccation creep viscosity (η_{dc}) and the basic creep consolidation parameter (κ) in the parameter range $\mu \in \mathcal{P} = [5\cdot 10^8, 5\cdot 10^{10}] \times [10^{-5}, 10^{-3}] \subset \mathbb{R}^2$. We consider a 7 by 7 training set Θ_{train}

Algorithm 5 POD strong-greedy process on projection errors (using ℓ_2-norm).

```
Require: Training set $\Theta_{\text{train}}$ of size $n_{\text{train}}$, $n_{\text{it}} \in \mathbb{N}_*$ the number of iterations
$\mathbf{Z} = []$, $\Theta_* = \emptyset$        ▷ Empty reduced basis and set of explored parameters
for $i_{\text{it}} \in \{1, ..., n_{\text{it}}\}$ do
    for $\mu \in \Theta_{\text{train}} \setminus \Theta_*$ do
        if $i_{\text{it}} == 1$ then
            Take the generalized coordinates $\forall k \in \{1, ..., K\}$:
                $$\mathbf{e}_\mu^{(k)} = \mathbf{u}_\mu^{\text{hf},(k)}$$
        else
            Compute error at every time step $\forall k \in \{1, ..., K\}$:
                $$\mathbf{e}_\mu^{(k)} = \mathbf{u}_\mu^{\text{hf},(k)} - \mathbf{Z}\mathbf{Z}^\top \mathbf{u}_\mu^{\text{hf},(k)}$$
        end if
        Compute the time-averaged error:
            $$E_\mu^{\text{avg}} = \sqrt{\frac{1}{T}\sum_{k=1}^{K} \Delta t_k \left\| \mathbf{e}_\mu^{(k)} \right\|_2^2} \Big/ \sqrt{\frac{1}{T}\sum_{k=1}^{K} \Delta t_k \left\| \mathbf{u}_\mu^{(k)} \right\|_2^2}$$
    end for
    Take the most poorly-approximated trajectory:
        $$\mu_* = \arg \max_{\mu \in \Theta_{\text{train}} \setminus \Theta_*} E_\mu^{\text{avg}} \tag{6.1}$$
    Update the set $\Theta_* = \Theta_* \cup \{\mu_*\}$    ▷ Can be formulated using indexes of parameters within $\Theta_{\text{train}}$
    if $i_{\text{it}} == 1$ then
        Compute reduced order basis:        ▷ Equation (2.27)
            $$\mathbf{Z}_u = \text{POD}\left(\{\Pi_{\mathcal{Z}^\perp} u_\mu\}_{k=1}^{K},\ (\cdot,\cdot)\ ,\ \varepsilon_{\text{POD},u}\right)$$
    else
            $$\mathbf{Z}_u = [\mathbf{Z}_u, \mathbf{Z}_{u,\text{new}}]\,, \quad \text{with} \quad \mathbf{Z}_{u,\text{new}} = \text{POD}\left(\{\Pi_{\mathcal{Z}^\perp} u_\mu\}_{k=1}^{K},\ (\cdot,\cdot)\ ,\ \varepsilon_{\text{POD},u}\right)$$
    end if
end for
```

and a 5 by 5 test set Θ_{test}. For both sets, parameters are logarithmically spaced in both directions. Figure 6.2 shows the behavior of (a) the normal force on a horizontal cable, and (b) the tangential and (c) vertical strains on the outer wall of the standard section of the containment building, for three distinct parameter values $\mu^{(i)} = (5.10^9, \kappa^{(i)})$, for $\kappa^{(i)} \in \{10^{-5}, 10^{-4}, 10^{-3}\}$, $i = 1, 2, 3$. As previsouly mentioned,"-E" notation indicates that the HF data are associated to the outer face of the structure. Figure 6.2 showcases the evolution of normal forces in the central horizontal cable (N_{H2}), the vertical (ε_{zz}) and the tangential (ε_{tt}) deformations on the outer surface of the geometry. Indeed, one can observe, for instance, that alterations in the consolidation parameter

κ exert an influence on the rate of decline of the diverse quantities.



(a) (b) (c)

Figure 6.2: Mechanical response of a NCB under external loading. Comparison of the quantities of interest computed for the two meshes of the standard section: (a) normal force on a horizontal cable, (b) tangential and (c) vertical strains on the outer wall of the standard section of the containment building.

An examination of these QoIs on both meshes illustrates that the evolution profiles are closely aligned. Figure 6.3 depicts the plots for a parameter at the center of the studied cartesian grid. The temporal evolutions are comparable, and the profiles overlap for all QoIs. Furthermore, this effect is substantiated through an analysis of errors across the training set. We introduce errors on the QoI by considering the temporal evolution:

$$E_\mu^{\max} = \frac{\max_{k=1,\dots,K} |q_\mu^{\mathrm{hf},(1)}(t^{(k)}) - q_\mu^{\mathrm{hf},(2)}(t^{(k)})|}{\max_{k=1,\dots,K} |q_\mu^{\mathrm{hf},(2)}(t^{(k)})|}, \quad E_\mu^{\mathrm{avg}} = \frac{\sum_{k=1}^{K} \Delta t^{(k)} |q_\mu^{\mathrm{hf},(1)}(t^{(k)}) - q_\mu^{\mathrm{hf},(2)}(t^{(k)})| \, dt}{\sum_{k=1}^{K} |q_\mu^{\mathrm{hf},(2)}(t^{(k)})|}. \tag{6.2}$$

where $q_\mu^{\mathrm{hf},(\mathrm{i})}$ denotes a given QoI for the mesh i. These two errors reflect the maximum error on the time trajectory ($E_\mu^{\max}$) and a time-averaged error over the entire simulated life of the structure (E_μ^{avg}). Table 6.1 shows the behavior of the maximum and average relative errors for the three QoIs of Figure 6.2. The statistical errors reveal a close equivalence in the results obtained from the two meshes in the context of this model problem.

	$\max_{\mu \in \mathcal{P}_{\mathrm{train}}} E_\mu^{\max}(\cdot)$	$E_\mu^{\max}(\cdot)/n_{\mathrm{train}}$	$\max_{\mu \in \mathcal{P}_{\mathrm{train}}} E_\mu^{\mathrm{avg}}(\cdot)$	$E_\mu^{\mathrm{avg}}(\cdot)/n_{\mathrm{train}}$
N_{H2}	$2.86 \cdot 10^{-2}$	$6.55 \cdot 10^{-4}$	$2.86 \cdot 10^{-2}$	$3.91 \cdot 10^{-5}$
$\varepsilon_{tt} - E$	$4.42 \cdot 10^{-2}$	$7.68 \cdot 10^{-3}$	$7.87 \cdot 10^{-3}$	$4.84 \cdot 10^{-3}$
$\varepsilon_{zz} - E$	$5.03 \cdot 10^{-2}$	$1.09 \cdot 10^{-2}$	$1.38 \cdot 10^{-2}$	$6.73 \cdot 10^{-3}$

Table 6.1: Mechanical response of a NCB under external loading. Computation of average (column 1 and 3) and maximum (column 2 and 4) errors over the training set for several errors on the quantities of interest: maximum error over all time steps (cf. Equation (6.2)).

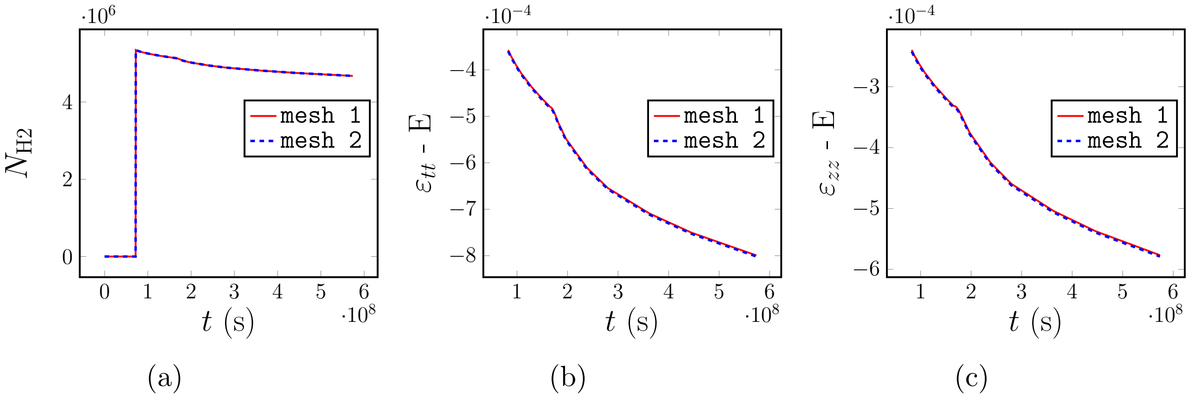


Figure 6.3: Mechanical response of a NCB under external loading. Comparison of the quantities of interest computed for the two meshes of the standard section: (a) normal force on a horizontal cable, (b) tangential and (c) vertical strains on the outer wall of the standard section of the containment building.

Table 6.2 offers a detailed breakdown of the costs associated with the HF solver across the training set for the two meshes. It is noteworthy that the wall-clock time for a complete HF simulation is around nine minutes for the coarse mesh and seventeen minutes for the refined mesh.

	mean	max	min	Q1	median	Q3
`mesh 1`	546.91	905.53	386.96	387.04	387.12	387.19
`mesh 2`	1034.07	1658.53	747.60	748.20	749.78	749.40

Table 6.2: HF CPU cost in seconds (s) for the HF simulations on the coarse (`mesh 1` - SS) and the refined mesh (`mesh 2` - SS).

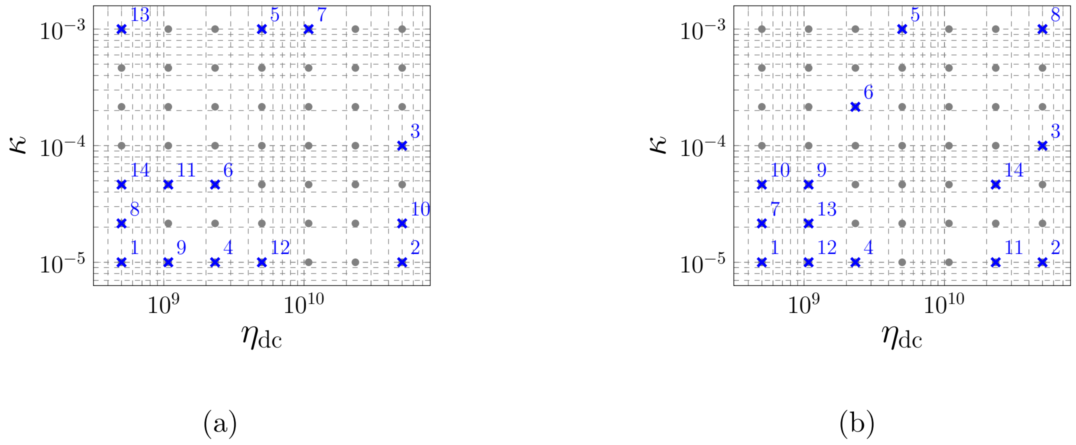
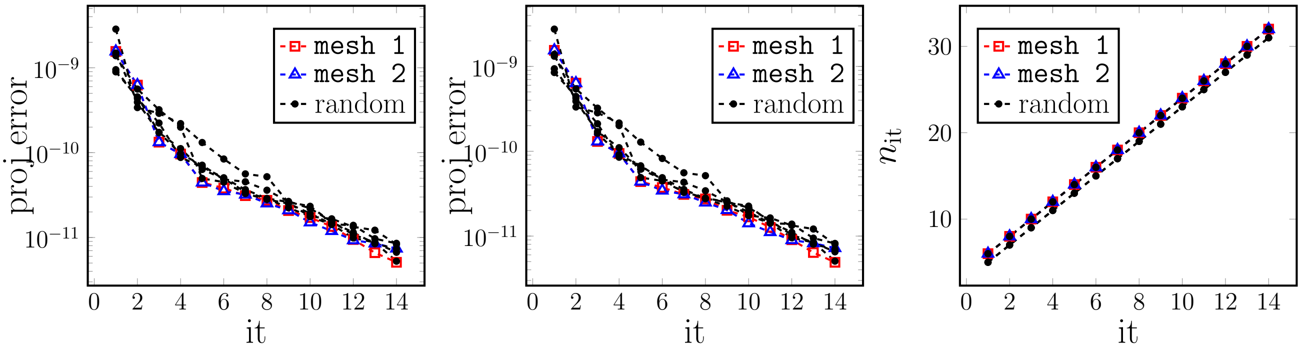


Figure 6.4: Parameters $\{\mu_j\}_j$ selected by Algorithm 5 for (a) the coarse mesh (`mesh 1` - SS) and (b) the refined mesh (`mesh 2` - SS).

The entirety of these comparisons, illustrated through this numerical case, demonstrates a scenario wherein a refined mesh, despite affording a more detailed approach to the problem, can be effectively approximated by a coarser counterpart. Within this example, solver calls on the coarser mesh are approximately half as computationally expensive. Employing the coarse mesh for sampling could potentially halve the computation time for an equivalent number of iterations. Figure 6.4 exhibits the parameters chosen by the algorithm for both the coarse (`mesh 1` - SS - cf. Figure 6.4a) and refined (`mesh 2` - SS - cf. Figure 6.4b) meshes. We can notice that a significant proportion of the sampled parameters is concentrated in the bottom-left corner of the parameter domain for both meshes. To assess the influence of the sampling strategy on performance, we quantify the projection error. Our objective is to compute the projection errors on a specified set (either test or train) for a given sampling:

$$E^{\text{proj}}_{\text{it}}(\Theta_\bullet) = \max_{\mu \in \Theta_\bullet} \frac{|||\mathbb{U}_\mu - \Pi_{\mathcal{Z}_{\text{it}}}\mathbb{U}_\mu|||}{|||\mathbb{U}_\mu|||}, \quad \bullet \in \{\text{train}, \text{test}\} \tag{6.3}$$

where $\Pi_{\mathcal{Z}_{\text{it}}}\mathbb{U}_\mu := \{\Pi_{\mathcal{Z}_{it}} u_\mu^{(k)}\}$ and $|||\mathbb{V}||| = \sqrt{\sum_{k=1}^{K} \Delta t^{(k)} \|v^{(k)}\|^2}$ is the discrete $L^2(0, T; \|\cdot\|)$ norm. For this case, we choose $\| \bullet \| = \| \bullet \|_2$ to be consistent with the norm used in the sampling strategy. This error is consistently computed on the finest mesh, treated as the reference solution. The performance, as measured through the projection error (cf Equation (6.3)), for the two samples depicted in the Figure 6.4 is presented in the Figure 6.5, considering both training and test sets. Remarkably, for this model problem, the choice of the sampling strategy exhibits minimal impact on performance. Nevertheless, even in this case, the greedy procedure based on the coarse mesh consistently outperforms random sampling. For this relatively regular problem, it appears judicious to employ a sampling strategy on a coarser mesh for parameter space exploration, as it does not seem to significantly degrade the efficacy of the strategy.



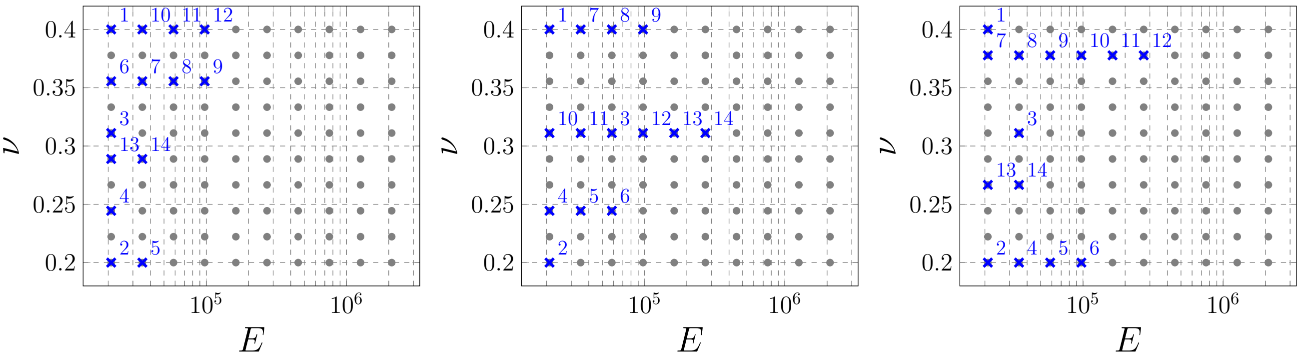
(a)

(b)

(c)

Figure 6.5: Mechanical response of a NCB under external loading. Behavior of the projection error $E^{\text{proj}}_{\text{it}}$ (6.3) for parameters selected by Algorithm 5 based on coarse (`mesh 1`) and fine data (`mesh 2`); comparison with random sampling: (a) performance on Θ_{test} (5×5); (b) performance on Θ_{train} (7×7); (c) behavior of the basis size n_{it}.

To showcase the potential of parameter sampling, we succinctly outline the exploration of the parametric space in the case of the elasto-plastic plate. Figure 6.6 depicts the parameter sampling for the three aforementioned meshes. Notably, the Poisson's coefficient ν exhibits a more significant impact than the Young's modulus E across all three meshes, which aligns with the inherent nonlinear nature of the solution with respect to ν, even within the elastic regime. Despite some differences, the sampling displays a similar structure in all three cases. This consistency across varying mesh fineness, given sufficiently close solutions, is further underscored by the second example. It highlights the capacity of this bi-fidelity strategy to capture nonlinear parameter



(a) (b) (c)

Figure 6.6: Parameters $\{\mu_j\}_j$ selected by Algorithm 5 for (a) the coarse mesh (`mesh 1`), (b) the intermediate mesh (`mesh 2`) and (c) the refined mesh (`mesh 3`).

dependencies, offering a potential avenue to reduce offline costs in greedy algorithms for nonlinear mechanics problems.

6.2.2 Incremental approaches for hyper-reduction procedures

In a second step, our emphasis is on expediting the construction of the EQ approach. The computation cost of building the EQ rule can escalate rapidly, especially with an increasing number of entries, due to the necessity of optimization algorithms. Our strategy involves developing an incremental approach to generate new EQ rules based on previously established rules. In this manner, one can expect to alleviate the computational burden in each iteration. More precisely, we aim to improve the efficiency of the hyper-reduction process when employing a hierarchical basis, where new modes are progressively added in each iteration. This is achieved by offering an incremental formulation of the ECSW approach. It is essential to note that the assumption of a hierarchical basis remains a fundamental aspect of the algorithm presented here.

Hierarchical bases in a greedy process imply a specific structure for the dictionary at the iteration n, denoted as $\mathbf{G}_n$. In this context, successive iterations mean adding rows to an existing matrix. Indeed, the EQ matrix $\mathbf{G}_n$ at iteration n satisfies:

$$\mathbf{G}_n = \begin{bmatrix} \mathbf{G}_{n-1} \\ \mathbf{G}_n^{\text{new}} \end{bmatrix} \tag{6.4}$$

where $\mathbf{G}_{n-1}$ is the old dictionnary (last greedy iteration) and $\mathbf{G}_n^{\text{new}}$ is the new lines added to the dictionnary. Readers can refer to Figure 2.3 to comprehend the structure of a given dictionnary. Each row represents information to be estimated, corresponding to a specific mode, a given parameter, and a time step. A suitable rearrangement readily leads back to an incremental building strategy of $\mathbf{G}_n$, following Equation (6.4). To provide a better grasp of the algorithm presented herein, we describe in greater detail the NNLS algorithm used in practice in this work [LH95]. The latter is detailed in Algorithm 6. The approach begins by taking a set of indices as input P_0, typically initialized with an empty set when there is no prior information, to initiate the process. Given the matrix $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_N] \in \mathbb{R}^{M\times N}$, the vector $\mathbf{x} \in \mathbb{R}^N$, we introduce the following notations: $\mathbf{G}(:, P) := [\mathbf{g}_{p_1}, \dots, \mathbf{g}_{p_m}] \in \mathbb{R}^{M\times m}$ and $\mathbf{x}(P) = \text{vec}((\mathbf{x})_{p_1}, \dots, (\mathbf{x})_{p_m}) \in \mathbb{R}^m$.

Moreover, we denote by $\#P$ the cardinality of the discrete set P, and we introduce the complement of P in $\{1, \ldots, N\}$ as $P^{\mathrm{c}} = \{1, \ldots, N\} \setminus P$. Given the vector $\mathbf{x} \in \mathbb{R}^N$ and the set of indices $\mathcal{I} \subset \{1, \ldots, N\}$, notation $[\alpha, i^\star] = \min_{i \in \mathcal{I}} (\mathbf{x})_i$ signifies that $\alpha = \min_{i \in \mathcal{I}} (\mathbf{x})_i$ and $i^\star \in \mathcal{I}$ realizes the minimum, $\alpha = (\mathbf{x})_{i^\star}$. The constant $\epsilon > 0$ is intended to avoid division by zero and is set to 2^{-1022}. The primary computational expense of Algorithm 6 arises from the repetitive solving of the least-squares problem at Line 11. Managing the computational cost necessitates restricting the number of iterations of the problem and, consequently, the number of least-squares problems to be addressed (output it in Algorithm 6). Although our study does not specifically delve into this aspect, it's worth noting that the size of successive problems varies with the number of iterations. Therefore, an optimal implementation of the algorithm would requires the utilization of solvers tailored to the system's size (such as QR or pseudo-inverse for instance). However, this falls beyond the scope of our current investigation. Our objective is to enhance the method's speed when using a given solver for Line 11 of Algorithm 6.

Algorithm 6 Active set method for the EQ-procedure (2.38).

Inputs: $\mathbf{G} \in \mathbf{R}^{M \times N}$, $\mathbf{b} \in \mathbb{R}^M$, $\delta > 0$, P_0.

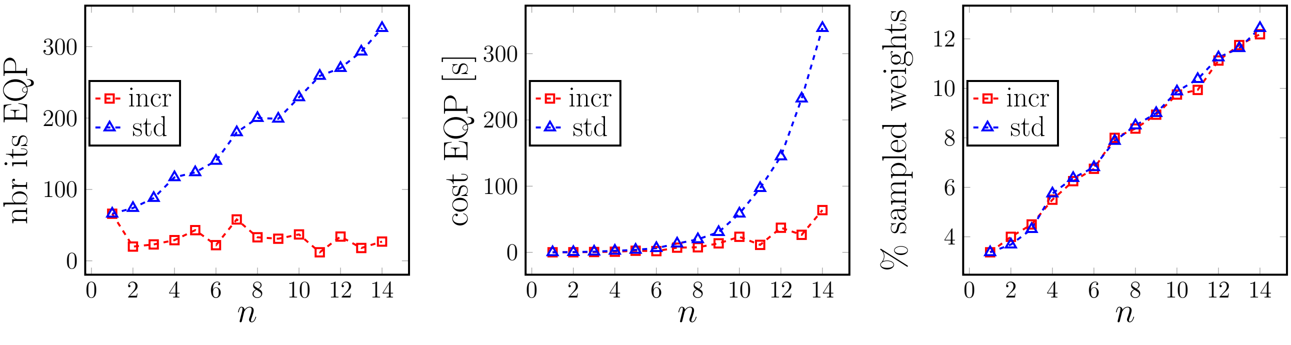
Output: $\boldsymbol{\rho}$ approximate solution to (2.38), it number of iterations to meet convergence criterion.

```
 1: Choose $\boldsymbol{\rho} = 0$, $\mathbf{w} = \mathbf{C}^\top \boldsymbol{\rho}$, $P = P_0$, it $= 0$.
 2: while true do
 3:     Compute $\mathbf{r} = \mathbf{G}(:, P)\mathbf{x}(P) - \mathbf{b}$.
 4:     if $\#P = N$ or $\|\mathbf{r}\|_2 \leq \delta \|\mathbf{b}\|_2$ then
 5:         break
 6:     end if
 7:     Set $i^\star = \arg\max_{j \notin P} (\mathbf{w})_j$.
 8:     Set $P = P \cup \{i^\star\}$.
 9:     while true do
10:         it = it + 1
11:         Define $\mathbf{z} \in \mathbb{R}^N$ s.t.$\mathbf{z}(P^{\mathrm{c}}) = 0$, $\mathbf{z}(P) = \mathbf{G}(:, P)^\dagger \mathbf{d}$
12:         if $\mathbf{z} \geq 0$ then
13:             Set $\mathbf{x} = \mathbf{z}$, $\mathbf{w} = \mathbf{G}^\top(\mathbf{b} - \mathbf{G}\mathbf{x})$
14:             break
15:         end if
16:         $\mathcal{I} = \{i \in \{1, \ldots, N\} : (\mathbf{z})_i < 0\}$.
17:         $[\alpha, i^\star] = \min_{i \in \mathcal{I}} \frac{(\mathbf{x})_i}{(\mathbf{x})_i - (\mathbf{z})_i + \epsilon}$.
18:         $P = P \setminus \{i^\star\}$.
19:         $\mathbf{x} = \mathbf{x} - \alpha(\mathbf{x} - \mathbf{z})$.
20:     end while
21: end while
```

As depicted in Equation (6.4), our data structure is designed so that new information pertains solely to the new entries $\mathbf{G}_n^{\text{new}}$. Initially, we possess a EQ rule capable of providing an estimate for the first block $\mathbf{G}_{n-1}$. Consequently, we advocate for initiating the optimization algorithm with a set founded on our prior knowledge, departing from an empty set. This set is formed as follows:

$$P = \left\{ i \in \{1, \ldots, N_{\mathrm{e}}\} : (\boldsymbol{\rho}^{\mathrm{eq},(n-1)})_i \neq 0 \right\}. \tag{6.5}$$

The ECSW algorithm is thus modified in an iterative way tailored for our greedy processes.



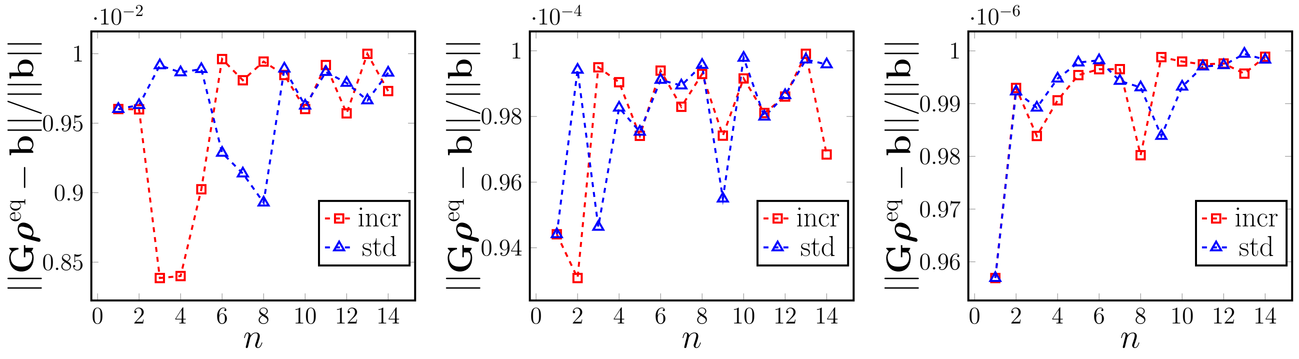
(a) (b) (c)

Figure 6.7: Progressive construction of the quadrature rule `mesh 2` for $\delta = 10^{-4}$: (a) number of iterations of the optimization algorithm; (b) CPU cost for the optimization algorithm; (c) percentage of sampled elements.

Figure 6.7 illustrates the performance evaluation of the EQ procedure in a test scenario focusing on the finer mesh (`mesh 2` - SS). The parameters $\{\mu^{\star,\mathrm{it}}\}_{\mathrm{it}=1}^{\mathrm{maxit}}$ are chosen using the POD-strong greedy Algorithm 5 based on HF results obtained from `mesh 1` (SS). Figures 6.7 (a)-(b)-(c) provide insights into the number of iterations necessary for Algorithm 6 to meet the convergence criterion, the computational cost, and the percentage of sampled elements, directly associated with online costs, for a given hyper-reduction tolerance $\delta = 10^{-4}$. Similar to the preceding test case, the results underscore the considerable reduction in NNLS iterations without compromising performance due to the incremental construction of the quadrature rule. Figure 6.8 shows the normalized residuals for several tolerances. These indicate the approximation quality for the integrals of interest. We can see that the convergence thresholds are similar for both incremental and classical algorithms.

Figure 6.9 examines the speedup of the incremental method for three chosen tolerances δ across different iterations of the greedy process. The observed trend reveals an increase in speedup with higher iteration counts, attributed to the diminishing percentage of new columns added during the it-th step in the matrix $\mathbf{G}_n$. Additionally, a notable acceleration in speedup is noted with decreasing tolerance, aligning with the increased number of iterations required by Algorithm 6 as δ is reduced.

To substantiate this insight, we propose an alternative procedure that slightly deviates from the conventional greedy strategy explored before. We sample parameters and consider constructing a reduced basis once for all obtained snapshots. Subsequently, we incrementally build the dictionary for each parameter using the previously employed approach. A noticeable distinction arises in the number of lines added at each greedy iteration. In the earlier case, with each iteration, the new lines corresponded to residuals computed for both new snapshots and newly obtained modes concerning previous snapshots, given the incremental nature of basis construction (case 1 in Figure 6.10). In the presented case, however, only terms associated with new snapshots are considered, as the basis was calculated only once at the outset (case 2 in Figure 6.10). Our earlier observation



(a) $\delta = 10^{-2}$ (b) $\delta = 10^{-4}$ (c) $\delta = 10^{-6}$

Figure 6.8: Normalized residuals for different hyper-reduction tolerances δ for the incremental and the standard approach on `mesh 2`

should imply more significant speedups in this scenario, as the percentage of added lines is lower at each iteration. Figure 6.10 provides a comparative analysis between the two numerical tests. We observe improved speedups with a less substantial addition of lines at each iteration, supporting our initial intuition. Employing this approach thus results in CPU time savings. Additionally, this observation suggests that a thoughtful consideration of matrix construction, in conjunction with method application, could contribute to accelerating computation time. However, we have not pursued a more in-depth analysis at this stage.

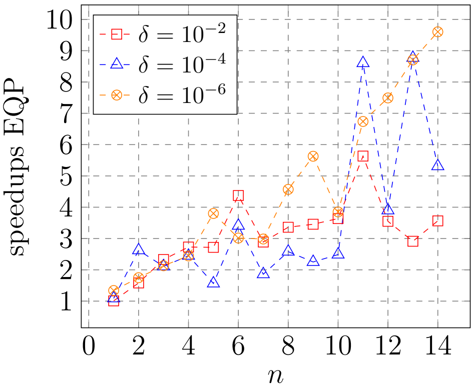


Figure 6.9: Speedups achieved when adopting the incremental approach 'incr' versus the standard approach 'std' for several hyper-reduction tolerances δ on `mesh 2` for a training set of size $7 \times 7 = 49$.

Much more, we wish to demonstrate that alternative numerical approaches could be employed to accelerate optimization algorithms. As an example, we introduce a second strategy that we have explored, which is less intrusive. Unlike the previous strategy, which involves modifying the algorithm with a non-empty initial active set, this second strategy introduces no alterations to the algorithm itself. Instead, it relies solely on the modification of the matrix and the right-hand side. The first step of this methodology is straightforward and is in the same spirit as before.: we aim to formulate a novel quadrature rule exclusively leveraging the newly acquired data: specifically, the new rows of the updated dictionary $(\mathbf{G}_{n-1})$. Consequently, we are equipped with two distinct

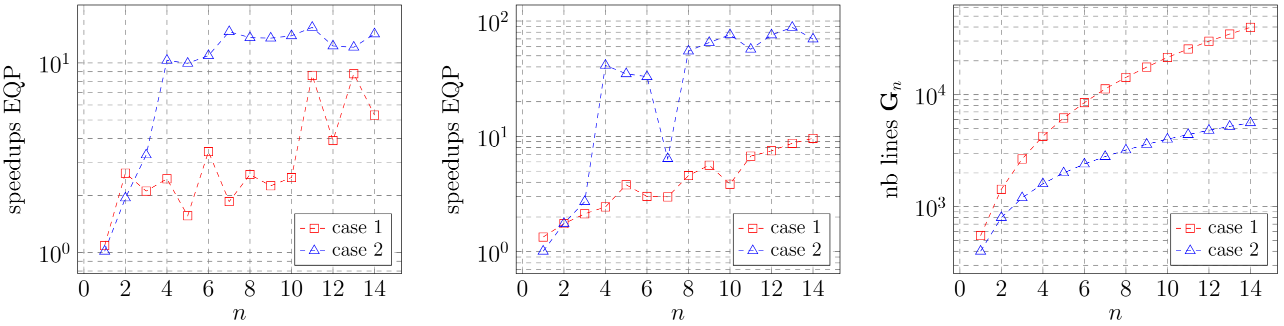


Figure 6.10: Comparison in speedups for the two approaches (case 1: pod-greedy vs case 2: already computed basis): speedups are presented for (a) a $\delta = 10^{-4}$ tolerance (b) $\delta = 10^{-6}$, while we present the number of dictionary lines at each iteration (c).

quadrature rules ($\boldsymbol{\rho}^{\text{eq}}_{n-1}$ and $\boldsymbol{\rho}^{\text{eq}}_{n,\text{new}}$). This scenario affords us the opportunity to delineate a set of indices ($\mathcal{I}$), defined as those indices where either of the two vectors exhibits a non-zero component. Subsequently, we can constrain the columns of the complete dictionary to this defined index set, thereby facilitating the resolution of a novel sparse approximation problem. Throughout this iterative process, it is essential to underscore that there is no need to redefine the second member : our overarching objective remains consistent since we aim at approximating the precise approximation of total integrals, spanning the entirety of the physical domain. All the steps in the algorithm for one iteration are summarized in Algorithm 7.

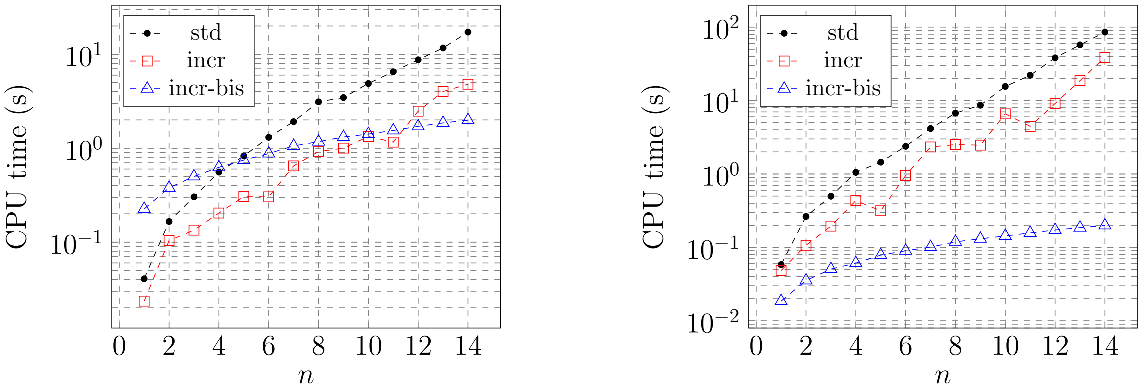
Algorithm 7 Loop for a direct incremental formulation.

```
Require: $\mathbf{G}_0 \in \mathbb{R}^{M_0 \times N_e}$, $\mathbf{G}_1 \in \mathbb{R}^{M_1 \times N_e}$, $\boldsymbol{\rho}^{\text{eq}}_0 \in \mathbb{R}^{N_e}$, $\delta$
1: Solve a EQ problem on the new dictionnary only:
       $\boldsymbol{\rho}^{\text{eq}}_1 = \text{EQ-solve}\left(\mathbf{G}_1,\ \mathbf{y}_1,\ \delta\right)$
2: Definition of a restricted set of indices:
       $\mathcal{I} = \mathcal{I}_0 \cup \mathcal{I}_1,\ \text{with} \left\{ \begin{array}{lcl} \mathcal{I}_0 & = & \{i \in \{1, ..., N_e\}, \text{such that } (\boldsymbol{\rho}^{\text{eq}}_0)_i \neq 0\} \\ \mathcal{I}_1 & = & \{i \in \{1, ..., N_e\}, \text{such that } (\boldsymbol{\rho}^{\text{eq}}_1)_i \neq 0\} \end{array} \right.$
3: Solve a EQ problem on the restriction of the dictionnary:
       $\boldsymbol{\rho}^{\text{eq}}_{\text{rest}} = \text{EQ-solve}\left(\mathbf{G}_{\text{rest}}, \mathbf{y}_{\text{rest}},\ \delta\right)$, where $\mathbf{G}_{\text{rest}} = \mathbf{G}[:, \mathcal{I}] \in \mathbb{R}^{(n_0+n_1)\times|\mathcal{I}|}$ and $\mathbf{y}_{\text{rest}} = [1, ..., 1]^\top \in \mathbb{R}^{|\mathcal{I}|}$
4: Definition of the new EQ rule:
       $\boldsymbol{\rho}^{\text{eq}}[\mathcal{I}] = \boldsymbol{\rho}^{\text{eq}}_{\text{rest}}$
   where $\boldsymbol{\rho}^{\text{eq}}[\mathcal{I}]$ this corresponds to the vector made up of the values of the initial vector ($\boldsymbol{\rho}^{\text{eq}}$) at the indices given by the set ($\mathcal{I}$).
```

In the approach detailed above, there is no guarantee that the two sets of indices obtained are disjoint. A second algorithm suggestion is to enforce that the two sets of indices are disjoint $\mathcal{I}_0 \cap \mathcal{I}_1 = \varnothing$. For this purpose, we define the complement of the first index set (known before any resolution) $\mathcal{I}_0^c = \{1, ..., N_e\} \setminus \mathcal{I}_0$, and we decide to solve a problem for a dictionnary whose columns indexes are restricted to the indices of the columns belonging to the complement:

$$\boldsymbol{\rho}_{1,\mathrm{rest}}^{\mathrm{eq}} = \text{EQ-solve}\left(\mathbf{G}_1[:,\mathcal{I}_0^c],\mathbf{y}_1,\delta\right), \quad \text{then} \quad \boldsymbol{\rho}_1^{\mathrm{eq}}[\mathcal{I}_0^c] = \boldsymbol{\rho}_{1,\mathrm{rest}}^{\mathrm{eq}}, \tag{6.6}$$

where $\mathbf{y}_1 = \mathbf{G}_1[:,\mathcal{I}_0^c]\cdot[1,...,1]^\top$. In this case, we could explore three distinct approaches: a first one, where the second member $\mathbf{y}$ is not modified to solve the first sub-problem; a second one, where a new second member is calculated, which means that we look for a quadrature rule that is only good on elements belonging to the complement of the reduced mesh initially constructed; and a third approach, where a new second member is calculated and the dictionary is renormalized. We choose the third option. Varying the initial algorithm in this manner helps stabilize the algorithm and, in particular, increases the quality of the solution obtained, bringing it closer to the quality achieved by the classical algorithm. We refrain from delving further into the details; however, additional information on other test cases can be found in Appendix F for the interested reader. We present some comparative results with the previous case for two relatively coarse tolerances on Figure 6.11. The results presented concern a case where the third approach is used, i.e. with an adaptation of the second member and a renormalization for step 3 of Algorithm 7. It is observed that we achieve rather similar speedups across the approaches. Nonetheless, when using this algorithm, convergence issues surfaced for overly restrictive tolerances, indicating the feasibility of deriving effective incremental optimization approaches to expedite computational costs. It is crucial to note that these findings constitute an initial stage of research. These approaches could be further enhanced or optimized to exert more control over the cost. In Appendix F, a connection between SVD and these approaches is presented, offering a research avenue that could be judiciously employed to control the size of considered dictionaries, for example. Furthermore, alternative resolution algorithms beyond the active set NNLS approach could also be explored.



(a) $\delta = 10^{-2}$

(b) $\delta = 10^{-3}$

Figure 6.11: CPU times for various incremental approaches versus the standard approach 'std' for several hyper-reduction tolerances δ on `mesh 2` - SS for a training set of size $7 \times 7 = 49$.

6.3 Conclusion

Effectively managing the computational demands during the offline training stage remains a significant hurdle in the application of model order reduction techniques to nonlinear, non-parametrically-affine problems. One potential avenue to mitigate this challenge is the use of adaptive sampling in the parameter domain through greedy methods, which could reduce the number of offline HF solves required to achieve the desired accuracy. However, it's worth noting that greedy methods are inherently sequential, introducing a non-negligible overhead that could potentially offset the advantages of adaptive sampling. To tackle these challenges, this study introduces two novel strategies to expedite greedy methods. The first strategy introduces a two-fidelity sampling approach aimed at reducing the number of computationally expensive greedy iterations. The second strategy ncorporates a warm start for the NNLS algorithm to determine the empirical quadrature rule. Furthermore, we point out that other ideas for improvements can be put forward to take advantage of the incremental aspect to speed up procedures. The numerical findings in this chapter underscore the effectiveness and applicability of our methods when addressing unsteady problems. Our numerical investigations reveal that adopting a sampling strategy based on coarse data yields performance that approaches optimality. This observation implies that multi-fidelity algorithms hold substantial promise in efficiently navigating the parameter domain during the training phase. Additionally, our findings highlight that incorporating a warm start for the NNLS algorithm results in a noteworthy reduction in computational costs without compromising performance.

The empirical results derived from this research inspire further exploration through theoretical and numerical investigations. One objective may be to develop both a priori and a posteriori indicators guiding the selection of mesh hierarchy when relying on multi-fidelity strategies. Secondly, we could extend the two key components of our formulation—progressive ROM generation and multi-fidelity sampling—to optimization problems. Specifically, in scenarios where the primary objective of model reduction is to estimate a targeted quantity of interest (goal-oriented model order reduction), we envision incorporating our formulation with adaptive techniques for optimization.

Chapter 7

Conclusion and perspectives

In this thesis, our research unfolds accross three distinct dimensions:

- first, we crafted projection-based ROMs with ECSW hyper-reduction tailored for nonlinear quasi-static mechanics problems entailing internal variables. The formulation of these ROMs is explicitly geared towards immediate practical integration into industrial-grade code `code_aster`;
- second, we devised a trust-region approach relying on ROMs as approximation models aimed at efficiently resolving calibration problems in a data assimilation framework for nonlinear solid mechanical problems;
- at last, our attention directed towards developing strategies that augment the applicability of iterative processes involving ROMs, with the overarching goal of fostering the pragmatic utilization of ROMs within industrial settings.

In the following paragraphs, we provide a comprehensive analysis of our findings, offering distinct perspectives on each facet of our research effort. This delineation facilitates a nuanced understanding of the contributions made in each area, enriching the overall discourse on the application of ROMs in the field of non-linear mechanics with internal variables using `code_aster`.

As a first step, we provided a numerical validation of the proposed ROM approach on a case of an elasto-plastic plate2. This test case matches a particular instance of a generic class of nonlinear mechanical problems with internal variables (single-modeling). The whole procedure delivered impressive computational cost improvements of $O(20-25)$ with relative prediction errors in the order of 10^{-3}. These results were carried out for a small number of parameters in order to conduct exhaustive numerical tests. Then, our ROM approach was also numerically validated for a more complexe problem, namely for prestressed concrete simulations (multi-modeling). The numerical approach proved extremely effective for a strong-greedy construction procedure with speedups above 10 even for a very coarse mesh. Therefore, we have a robust approach, implemented in an industrial setting, for three-dimensional meshes capable of handling a multi-modeling framework. Additionally, the devised ROMs facilitate the straightforward construction of quantities of interest crucial for engineering analyses. These quantities fall into two categories: first, the normal forces in the bars, employed for estimating leakage rates. Second, ROMs assist in generating localized deformations at the sensor level, which are valuable for recalibration procedures through data assimilation. In conclusion, the ROM approach aligns well with the quasi-static non-linear problems integrated into the industrial code `code_aster` and the engineers' needs. When applied to the specific case under consideration, ROM not only achieves substantial computational time savings but also ensures a certified quality of approximation for both the primal variable (displacement) and the quantities of interest.

In a second step, we implemented a trust-region approach with ROM as approximation model. First, we demonstrated the relevance of using this approach for calibration through data assimilation, which can be viewed as a PDE-constrained optimization problem. We have demonstrated the feasibility and advantages of this approach through numerical illustrations involving twin experiments in both linear and non-linear elasticity using `Fenics`. These experiments covered global observations, such as the entire displacement field, as well as local operators. This expansion of existing methodologies has allowed us to tailor the approach to our specific focus on parameter calibration. Furthermore, we have even shown that the resolution approach is compatible with various resolution processes (gradient-free, gradient-approximated) and compatible with using an industrial-grade simulation tool `code_aster`.

Finally, we introduced methods to expedite iterative processes for resource-intensive strategies, delineated into two components: first, a bi-fidelity approach for systematically sampling the parametric space to construct the reduced model; and second, an incremental strategy to reduce the offline computational cost for hyper-reduction. We proposed and numerically validated these approaches to enhance the efficiency of resource-intensive processes during ROM construction.

In summary, this research led us to formulate cutting-edge methodologies tailored to the industrial context. These methodologies address the challenges posed by engineers, including the analysis of quantities of interest and calibration problems. The development has taken place within an environment relying on high-quality industrial code. Furthermore, we have explored approaches compatible with this environment, such as implementing an ECSW approach through elementary integral reconstruction outside the code, devising an error indicator for single-modeling cases, and exploring alternatives to adjoint methods for optimization problems. Moreover, the emphasis of this work is on practical applicability. This has led us to focus on on-the-fly ROM construction, perfectly suited to practical application, and on acceleration techniques to minimize offline construction costs.

The potential for ongoing research is extensive, spanning across the three specific directions explored in this study. First, the approach in the multi-modeling case has not been tested on a weak-greedy approach. Thus, the approach in this case relies on comparison with known HF snapshots. This leads to significant offline computation costs, since it requires a priori knowledge of these solutions. This is a particular limitation when scaling up. Previous efforts have focused on the construction of low-cost a posteriori error indicators. Since the efficiency of these indicators in steering greedy search (within a weak-greedy context) has been demonstrated for problems featuring internal variables, the derivation and implementation of efficient error indicators would be a key point for computational efficiency. Once this door is opened, a second challenge arises, which is dealing with the increase in dimensionality of the parametric space. Indeed, the constitutive equations involved in these studies require several dozen parameters, and the dimensionality increase could be a scientific hurdle to maintaining the validity of the ROM across the entire parametric range. Ultimately, our efficient ROM approach could lead to the exploration into more intricate applications, particularly within the realm of engineering problems. The mesh configurations utilized here, in fact, embody a historical compromise aligned with the computational capacities accessible at a specific point in time. Given the advancements achieved in this thesis, a logical progression would involve contemplating finer meshes or more intricate geometries, extending beyond the confines of the concrete material representative volume.

Second, many avenues of work remain to be explored for the calibration procedure, both from a theoretical point of view and in terms of applications for engineers. In terms of applications, previous efforts coupled with an implementation with `code_aster` indicate that it is now feasible to provide a calibration model based on ROMs for the standard section of a nuclear containment building. The scope of work in this direction may be broken down into three stages: first, a simple application of the method to the industrial case, by performing a recalibration on the total displacement field; second, an application to the case with the observation operators used in practice by engineers; and finally, a recalibration on real data. This three-stage work plan is intended to enable projection-based ROMs for optimization to be used on a real-world engineering problem. Various issues could arise during these studies, such as the relevance of the choice of modes to the observation operators used, or the problem's sensitivity to measurement noise in real-world applications. A second challenge would be to develop a more mathematically robust implementation for trust-region algorithm. From the optimization algorithm implementation point of view, our primary focus was to apply the method to industrial cases. A finer-grained analysis of the solvers would certainly enable an error analysis with greater control. In terms of a more theoretical line of research, a mathematical analysis to account for error indicators, in order to perform error-aware trust-region or adapt the reduced model in another way, would provide more error control. This would make this method more valuable, especially in competition with other meta-modeling processes where error analysis is more complicated.

Finally, when devising approach for accelerating iteratives processes involving ROMS, one should also draw attention on optimization processess. Indeed, within the scope of this thesis, calibration problems are also investigated. Building upon the bi-fidelity approach for parameter sampling, a multi-fidelity approach could be explored to offer a robust numerical strategy for calibration problems in a data assimilation framework. This could complement the perspectives presented for trust-region algorithms, particularly in terms of error control. Another avenue for exploration would involve concentrating on the construction of the reduced basis for optimization processes. While the on-the-fly construction proposed here concatenates successive iterates to form a reduced basis, it is worth noting that, during an optimization process, the final iterates are closer to the solution than the initial point. Effectively managing the ROM could involve assigning relative importance to the various snapshots incorporated. Thus, snapshots could be weighted to establish a sliding basis.

Appendix A

Implementation with `code_aster` and `mordicus`

In this appendix, we provide details on the numerical implementation of algorithms within our development environment, namely in our industrial-grade FE code `code_aster`, and with the `Python` package `mordicus`. First, we detail the interactions between those two softwares. Second, we present the details of the Newton algorithms applied within the HF code (with `code_aster`). Furthermore, in the context of the work presented in this thesis, we demonstrate an application of the methodology to cases with homogeneous boundary conditions. We extend the approach to cases with non-homogeneous conditions and outline the necessary modifications in the methodology or FE solver to handle these scenarios. This illustrates that such boundary conditions do not pose limitations to our approach. Finally, we delve into the specifics of the hyper-reduction approach for our reduction methodology: we outline the method for constructing our dictionaries in practice and provide details on building a reduced mesh within the framework of `code_aster`.

In this section, we detail the interactions between `code_aster`, and with the `Python` package `mordicus`. The industrial-grade code `code_aster` functions by taking command files as input, specifying the constitutives equations, numerical models and algorithms to be employed, along with mesh files or results serving as an initial point (such as a thermal calculation for a thermo-mechanical calculation using weak coupling). Additionally, a link file is utilized to establish connections between various objects. On the other hand, `mordicus` is designed to be software-agnostic when it comes to input. Reader functions are established for the specific software in use, for instance, `code_aster`, enabling the export of desired data (e.g., snapshots). This ensures that the data structure remains independent of the initial software. Subsequent ROM procedures, such as reduced bases generation or hyper-reduction, can be applied to this structure, and files are then exported for compatibility with the reference code in use.

The developments had to be articulated on several levels. In the case of `code_aster`, extensive development and adaptation were necessary to enable the existing software to accommodate a reduced solver for hyper-reduced models utilizing the ECSW approach. As for `mordicus`, we had to add the functionalities of reduced base construction, hyper-reduction and building a reduced mesh. Furthermore, it's noteworthy that all the presented test cases are encapsulated within `Python` files, each invoking the diverse functionnalities of `mordicus`, including file reads, external calls to the FE solver, construction of ROMs, and generation of reduced solutions generated through `code$aster`.

A.1 Newton solver

In this section, we provide details on the numerical procedure used for solving nonlinear systems with dualisation of boundary conditions in the `code_aster` framework. For this purpose, we first discuss the procedure used when the kinematic conditions are handled by Dirichlet elimination, before introducing the dualization of the boundary conditions and the stopping criteria considered.

A.1.1 No dualisation of the boundary conditions

We focus on looking for the k-th timestep solution . The resolution is performed by a Newton-Raphson type algorithm, which is an incremental algorithm. The iterative process is driven by the search for a solution at each iteration according to the knowledge at the previous iteration:

$$\mathbf{u}_{\theta+1}^{(k)} = \mathbf{u}_{\theta}^{(k)} + \Delta\mathbf{u}_{\theta}^{(k)}$$

The iterate is computed from the solution of the linear system, expressed with the Jacobian matrix (also called tangent matrix in mechanics) evaluated in $\mathbf{u}_{\theta}^{(k)}$:

$$\mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_{\theta+1}^{(k)}\right) \approx \mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_{\theta}^{(k)}\right) + \mathbf{K}_{\theta}^{(k)} \cdot \Delta\mathbf{u}_{\theta}^{(k)} = \mathbf{0}, \quad \text{with} \quad \mathbf{K}_{\theta}^{(k)} = \frac{\mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\left(\mathbf{u}_{\theta}^{(k)}\right)$$

A.1.2 Dualization of the boundary conditions

For the dualisation of constraints, we must investigate a new increment in displacement and in terms of Lagrange multipliers:

$$\left\{\begin{array}{rcl} \mathbf{u}_{\theta+1}^{(k)} & = & \mathbf{u}_{\theta}^{(k)} + \Delta\mathbf{u}_{\theta}^{(k)} \\ \boldsymbol{\lambda}_{\theta+1}^{(k)} & = & \boldsymbol{\lambda}_{\theta}^{(k)} + \Delta\boldsymbol{\lambda}_{\theta}^{(k)} \end{array}\right. \tag{A.1}$$

The task is hence to solve the following nonlinear system:

$$\left\{\begin{array}{rcl} \mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_{\theta+1}^{(k)}\right) + \mathbf{B}^T\boldsymbol{\lambda}_{\theta+1}^{(k)} & = & \mathbf{0} \\ \mathbf{B}\mathbf{u}_{\theta+1}^{(k)} & = & \mathbf{u}_{d}^{(k)} \end{array}\right. \tag{A.2}$$

Using a linearization analogous to the equation, and exploit the linearity of the operator associated with the kinematic conditions, the assembled discretized system (for one θ iteration) is decomposed as:

$$\left\{\begin{array}{rcl} \mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_{\theta}^{(k)}\right) + \mathbf{K}_{\theta}^{(k)} \cdot \Delta\mathbf{u}_{\theta}^{(k)} + \mathbf{B}^T\boldsymbol{\lambda}_{\theta}^{(k)} + \mathbf{B}^T\Delta\boldsymbol{\lambda}_{\theta}^{(k)} & = & \mathbf{0} \\ \mathbf{B}\mathbf{u}_{\theta}^{(k)} + \mathbf{B}\Delta\mathbf{u}_{\theta}^{(k)} & = & \mathbf{u}_{d}^{(k)} \end{array}\right. \tag{A.3}$$

which leads to the following saddle-point problem:

$$\begin{bmatrix} \mathbf{K}_{\theta}^{(k)} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_{\theta}^{(k)} \\ \Delta\boldsymbol{\lambda}_{\theta}^{(k)} \end{bmatrix} = \begin{bmatrix} -\mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_{\theta}^{(k)}\right) - \mathbf{B}^T\boldsymbol{\lambda}_{\theta}^{(k)} \\ \mathbf{u}_{d}^{(k)} - \mathbf{B}\mathbf{u}_{\theta}^{(k)} \end{bmatrix} \tag{A.4}$$

A.1.3 Stopping criterium

Under the philosophy of the formulations in `code_aster`, the internal contributions (work of internal forces) and external contributions (forces applied to the system) are evaluated separately in the assembled residue:

$$\mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}^{(k)}\right) = \mathbf{F}^{\mathrm{int},(k)}\left(\mathbf{u}^{(k)}\right) - \mathbf{F}^{\mathrm{ext},(k)} \tag{A.5}$$

Different criteria are available in `code_aster`. The reader may refer to the code documentation for more details. Our choice is a relative criterion defined as follows:

$$\frac{\left\|\mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_\theta^{(k)}\right) + \mathbf{B}^T\boldsymbol{\lambda}_\theta^{(k)}\right\|_\infty}{\left\|\mathbf{B}^T\boldsymbol{\lambda}_\theta^{(k)} - \mathbf{F}^{\mathrm{ext},(k)}\right\|_\infty} \leq \varepsilon_{\mathrm{newt}} \tag{A.6}$$

The vector $\mathbf{B}^T\boldsymbol{\lambda}_\theta^{(k)}$ can be interpreted physically as the opposite of the support reactions at the nodes where the conditions are dualised. The convergence criterion can be seen as a process of normalizing the residual calculated at a given iteration with respect to the forces exerted on the system at that iteration (external forces and support reactions).

A.2 Model order reduction when using inhomogeneous Dirichlet conditions

A.2.1 Affine decomposition

We consider the situation where the boundary conditions are non-homogeneous. We remain within the framework where the kinematic link matrix is independent of the parameter or displacement vector, which is a fundamental assumption of our work. We shall assume BCs in the following form:

$$\mathbf{B}\mathbf{u}_\mu^{(k)} = \mathbf{u}_d^{(k)}, \quad \forall \mu \in \mathcal{P}$$

where $\mathbf{u}_d^{(k)}$, $\forall k \in \{1, ..., K\}$ are the BCs vectors used. In order to build a ROM, we need to rely on a sequence of lifting vectors $\overline{\mathbf{u}}^{(k)}$, $\forall k \in \{1, ..., K\}$. We define this sequence as follows:

$$\mathbf{B}\overline{\mathbf{u}}^{(k)} = \mathbf{u}_d^{(k)}$$

The reduced basis would then be produced from the snapshots by subtracting the lift terms:

$$\mathbf{Z}_u = \mathrm{POD}\left\{\left\{u_\mu^{(k)} - \overline{u}^{(k)}\right\}_{k=1}^{K},\ (.,.),\ \varepsilon_{\mathrm{POD,u}}\right\}$$

Thus, we obtain modes that meet homogeneous conditions $\forall n \in \{1, ..., N_u\}$ $\mathbf{B}\boldsymbol{\zeta}_{u,n} = 0$. We seek the solution as a linear combination of modes:

$$\widehat{\mathbf{u}}_\mu^{(k)} = \overline{\mathbf{u}}^{(k)} + \sum_{n=1}^{N_u}\left(\widehat{\alpha}_\mu^{(k)}\right)_n \boldsymbol{\zeta}_{u,n} = \overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u\widehat{\boldsymbol{\alpha}}_\mu^{(k)} \tag{A.7}$$

where $\widehat{\boldsymbol{\alpha}}_{u,\mu}^{(k)} \in \mathbb{R}^{N_u}$ are referred to as generalized coordinates and $\mathcal{Z}_{N_u} = \mathrm{span}\left\{\zeta_{u,n}\right\}$ is the primal reduced space. Therefore, the residual projection holds the same properties as in the homogeneous case:

$$\mathbf{Z}_u^T \cdot \left[\mathbf{R}_\mu\left(\widehat{\mathbf{u}}_\mu^{(k)}\right) + \mathbf{B}^T \cdot \boldsymbol{\lambda}_\mu^{(k)}\right] = 0 \quad \Rightarrow \quad \mathbf{Z}_u^T \cdot \mathbf{R}_\mu\left(\widehat{\mathbf{u}}_\mu^{(k)}\right) = 0$$

A.2.2 Reduced Newton iteration for inhomogeneous boundary conditions

We present here the details of the calculation steps for the reduced solver in this case. Reduced Newton iterations can be formulated as follows:

$$\mathbf{Z}_u^T \cdot \frac{\partial \mathbf{R}_\mu}{\partial \mathbf{u}}\left(\widehat{\mathbf{u}}_{\mu,\theta}^{(k)}\right) \cdot \Delta\widehat{\mathbf{u}}_{\mu,\theta}^{(k)} = -\mathbf{Z}_u^T \cdot \mathbf{R}_\mu\left(\widehat{\mathbf{u}}_{\mu,\theta}^{(k)}\right) \tag{A.8}$$

for a given $\mu \in \mathcal{P}$ and a given $k \in \{1, ..., K\}$. We rely on the affine decomposition presented above (see Equation (A.7)):

$$\mathbf{Z}_u^T \cdot \frac{\partial \mathbf{R}_\mu}{\partial \mathbf{u}}\left(\overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)}\right) \cdot \Delta\widehat{\mathbf{u}}_{\mu,\theta}^{(k)} = -\mathbf{Z}_u^T \cdot \mathbf{R}_\mu\left(\overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)}\right)$$

We introduce the formulation with reduced coordinates in reduced Newton iterations:

$$\underbrace{\mathbf{Z}_u^T \cdot \frac{\partial \mathbf{R}_\mu}{\partial \mathbf{u}}\left(\overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)}\right) \cdot \mathbf{Z}_u}_{=\mathbf{K}_{\mu,\theta}^{(k),\mathrm{r}}} \Delta\widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)} = -\mathbf{Z}_u^T \cdot \mathbf{R}_\mu\left(\overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)}\right)$$

which leads to:

$$\mathbf{K}_{\mu,\theta}^{(k),\mathrm{r}} \cdot \Delta\widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)} = -\mathbf{Z}_u^T \cdot \mathbf{R}_\mu\left(\overline{\mathbf{u}}^{(k)} + \mathbf{Z}_u \widehat{\boldsymbol{\alpha}}_{\mu,\theta}^{(k)}\right)$$

The main difference with the previous case during the online phase is simply to add the lifting terms during Newton iterations, so as to calculate the residuals and Jacobian matrices at the relevant points.

A.2.3 Implementation in the industrial code

Generic remarks

It is worth highlighting that the required changes to the industrial solver would be the following: the calculation of a lifting term satisfying the boundary conditions, and the addition of this lifting term in the residual (and the Jacobian matrix) evaluations at each Newton iteration. However, these implementations have not been carried out in the industrial code at our disposal, as this sort of boundary condition is an extension of the case considered in our work.

Simplified framework, initial guess and lifting

In the following, we describe a special case of BCs that do not require modifications to the heart of the code. We show that even with the current state of the art of industrial code, it is possible to increase the complexity of simulated boundary conditions, without the need for further development. Specifically, we consider a case where the Dirichlet data is common to all time steps:

$$\forall k \in [\![1, K]\!], \qquad \mathbf{u}_d^{(k)} = \mathbf{u}_d$$

We detail the expression of a Newton iteration as a function of the previous iterations. This is written as follows. As a result, we can work:

$$\begin{aligned}
\mathbf{u}_{\mu,\theta+1}^{(k)} &= \mathbf{u}_{\mu,\theta}^{(k)} + \Delta\mathbf{u}_{\mu,\theta}^{(k)} \\
&= \sum_{\gamma=1}^{\theta} \Delta\mathbf{u}_{\mu,\gamma}^{(k)} + \mathbf{u}_{\mu,0}^{(k)} \\
&= \sum_{\gamma=1}^{\theta} \mathbf{Z}_u \cdot \Delta\boldsymbol{\alpha}_{\mu,\gamma}^{(k)} + \mathbf{u}_{\mu,0}^{(k)} \\
&= \mathbf{Z}_u \cdot \underbrace{\left(\sum_{\gamma=1}^{\theta} \Delta\boldsymbol{\alpha}_{\mu,\gamma}^{(k)} \right)}_{=\boldsymbol{\alpha}_{\mu,\theta+1}^{(k)}} + \mathbf{u}_{\mu,0}^{(k)}
\end{aligned}$$

If we take the initial guess as the lifting, we have the expression we need

$$\mathbf{u}_{\mu,\theta+1}^{(k)} = \mathbf{Z}_u \cdot \boldsymbol{\alpha}_{\mu,\theta+1}^{(k)} + \mathbf{u}_{\mu,0}^{(k)} \quad \Rightarrow \quad \mathbf{u}_{\mu,\theta+1}^{(k)} = \mathbf{Z}_u \cdot \boldsymbol{\alpha}_{\mu,\theta+1}^{(k)} + \overline{\mathbf{u}}^{(k)} \tag{A.9}$$

A.3 Details on the hyper-reduction implementation

A.3.1 Computing a reduced mesh in `code_aster`

In order to comprehend the construction of a reduced mesh in `code_aster`, it is necessary to take some time to comprehend the structure of the utilized mesh and the process for making calls to conduct a numerical simulation. The FE code used takes a '.comm' command file as input, which reads input files, including the mesh saved in a '.med' format. MED serves as the data exchange model of SALOME, offering a standardized representation of meshes and result fields independent of the simulated physics.

Meshes are constructed to have a set of nodes defined throughout the entire mesh, followed by several mesh levels. For instance, in a three-dimensional mesh, there is a mesh containing the 3D meshes, then a 2D mesh to account for volumetric loads, and possibly a 1D mesh if required. These three mesh levels are defined on the same node field (common to all three and defining the overall mesh), but the integration points differ, each being specific to its mesh level.

Regarding the command file, it is essential to define how to impose loads. In practice, loads (volumetric or surface types) can be applied at the node (`GROUP_NO` commande) or element level (`GROUP_MA` commande). It is noteworthy that there are cases where both forms are equivalent. For example, certain loads on elements may involve applying nodal forces to all nodes of the elements in a group. Thus, the concept of node and element groups emerges, utilized in mesh construction to have groups of elements (or nodes) to which the same load or boundary condition is applied.

(a) Géométries et noeuds du maillage exemple

(b) Visualisation des groupes de noeuds et de mailles

Figure A.1: Example of a mesh for a rectangle (representing, for instance, a fixed-end beam under tension), featuring a node group (fixed support), an element group (elements where the tensile force is applied), and the associated node group (nodes associated with these elements). This mesh is a simple representation of a mesh for a beam under tensile load, for example

Within our framework, for simplicity in practical implementation, we have opted not to continually rewrite command and call files to avoid an excessive proliferation of written files. Consequently, our advancements necessitate the presence of all initially required node and mesh groups in the final mesh. It is worth noting that this requirement is specific to the practical construction of meshes in our context and may not be essential for a broader application of the methodology. However, it is crucial to consider when visualizing the reduced meshes presented in our work.

For each constructed test case, we can establish a set of required node groups and mesh groups to facilitate the computation. In the illustrated example, maintaining at least one green node and one magenta mesh is imperative to accurately apply BCs (in the initial '.comm' even though we do not apply them in pratice) and exert tensile forces. Consequently, an initial corrective step is taken to rectify the reduced mesh by adding elements in cases where mesh groups or nodes are absent. In instances where a necessary mesh group (or node) is absent in the initial construction, an element (or an element with vertices belonging to the node group) is randomly selected and incorporated into the reduced mesh. Additionally, in the case of a multi-level mesh like the one depicted here, a second correction step is essential for completing the mesh construction. If a 2d mesh is chosen, the adjacent 3d mesh is appended to the reduced mesh.

A.3.2 Dictionnary construction

Formulation

We resume the example of a solution reproduction problem for a single-modeling case. We describe more precisely the hyper-reduction process used in our methodology. We keep the same notations as previously introduced. In such a scenario, we have K HF snapshot (displacements and stresses) and N_u primal modes at our disposal. We hence have $n_{\mathrm{int}} = K \times N_u$ *manifold accuracy constraints* to fulfill:

$$(\mathbf{G})_{\mathrm{lines}(n,k),\ q} = \mathcal{R}_q^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) \quad \text{and} \quad (\mathbf{y})_{\mathrm{lines}(n,k)} = \mathcal{R}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) \tag{A.10}$$

where $\mathbf{G} \in \mathbb{R}^{n_{\mathrm{int}} \times N_e}$ and $\mathbf{y} \in \mathbb{R}^{n_{\mathrm{int}}}$ and lines : $(\mathrm{k}, \mathrm{n}) \in \mathbb{R}^{\mathrm{K} \times \mathrm{N_u}} \to \mathbb{R}^{\mathrm{n_{int}}}$ a bijection used to have a unique numerotation of rows (set by the way we build the dictionnary). The last row of the dictionnary is set in order to fulfill the *constant-function constraint*:

$$(\mathbf{G})_{n_{\mathrm{int}}+1,\ q} = |\mathtt{D}_q|\,, \quad \text{and} \quad (\mathbf{y})_{n_{\mathrm{int}}+1} = |\Omega| \tag{A.11}$$

Separation of integrals

As we restrict ourselves to a single-mesh study, we have only volumic forces applied to the system. From a practical viewpoint, adding directly the residuals can load to numerical instabilities. Indeed, if the probelm is well represented by a single mode, $\zeta_{u,n*}$, we can have:

$$\mathcal{R}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) \approx \mathbf{0}$$

To tackle this issue, we chose to split the residual in two contributions: one for the internal forces and the other for the external forces. Such an implementation is consistent with `code_aster` discrete formulation. The residuals can be expressed thanks to the variationnal form as:

$$\mathcal{R}_q^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) = \int_{\Omega_q} \sigma^{(k)} : \nabla_s \zeta_{u,n}\, dx - \int_{\Omega_q} f_v \cdot \zeta_{u,n} dx \tag{A.12}$$

We can then define the contributions:

$$\left\{\begin{array}{lcl} \mathcal{R}_q^{\sigma,\mathrm{hf,int}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) & = & \int_{\Omega_q} \sigma^{(k)} : \nabla_s \zeta_{u,n}\, dx \\ \mathcal{R}_q^{\sigma,\mathrm{hf,ext}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) & = & \int_{\Omega_q} f_v \cdot \zeta_{u,n} dx \end{array}\right. \tag{A.13}$$

With this formulation, we have $n_{\mathrm{lin}} = (K+1) \times N_u$ and $\mathbf{G}$ and $\mathbf{y}$ are modified accordingly:

$$(\mathbf{G})_{\mathrm{lines}(n,k,*),\ \mathrm{q}} = \mathcal{R}_q^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) \quad \text{and} \quad (\mathbf{y})_{\mathrm{lines}(n,k,*)} = \mathcal{R}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right)$$

Normalization

A challenge related to the orders of magnitude arises in the optimization problem construction. Indeed, we have lines related to volume constraints while others are related to internal or external forces. Since the algorithms convergence criteria used are designed on the residuals (in the sense of optimisation, i.e. $\|\mathbf{G}\boldsymbol{\rho} - \mathbf{y}\|_*$), it is likely that some constraints are 'overlooked' because of the differences in order of magnitude. To ensure a good behaviour of our strategy, we normalize the whole dictionary to have an addimensionalized problem:

$$(\mathbf{G})_{\mathrm{lines}(n,k,*),\ \mathrm{q}} = \frac{\mathcal{R}_q^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right)}{\mathcal{R}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}^{(k)},\ \mathbf{E}_q^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right)} \quad \text{and} \quad (\mathbf{G})_{n_{\mathrm{int}}+1,\ q} = \frac{|\mathtt{D}_q|}{|\Omega|} \tag{A.14}$$

Thus, the second member consists only of a unitary vector:

$$(\mathbf{y})_{\mathrm{lines}(n,k,*)} = 1, \quad \text{and} \quad (\mathbf{y})_{n_{\mathrm{int}}+1} = 1 \tag{A.15}$$

This approach is well suited to industrial codes that are not necessarily designed to have dimensionless formulations.

A.3.3 Example of visualisation of a dictionnary

In this section, we provide a visualization of the dictionary. We consider a dictionary of size $\mathbf{G} \in \mathbb{R}^{(n_{\mathrm{train}} \times K \times N_u) \times N_{\mathrm{e}}}$. In our framework, the dictionnary is of the following form:

$$
\mathbf{G} = \left[\begin{array}{c:c:c}
\mathcal{R}_1^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(1)},\ \mathbf{E}_1^{\mathrm{no}}\boldsymbol{\zeta}_{u,1}\right) & \dots & \mathcal{R}_{N_\mathrm{e}}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_{N_\mathrm{e}}^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(1)},\ \mathbf{E}_{N_\mathrm{e}}^{\mathrm{no}}\boldsymbol{\zeta}_{u,1}\right) \\
\dots & \dots & \dots \\
\mathcal{R}_1^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(K)},\ \mathbf{E}_1^{\mathrm{no}}\boldsymbol{\zeta}_{u,1}\right) & \dots & \mathcal{R}_{N_\mathrm{e}}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_{N_\mathrm{e}}^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(K)},\ \mathbf{E}_{N_\mathrm{e}}^{\mathrm{no}}\boldsymbol{\zeta}_{u,1}\right) \\
\mathcal{R}_1^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(1)},\ \mathbf{E}_1^{\mathrm{no}}\boldsymbol{\zeta}_{u,2}\right) & \dots & \mathcal{R}_{N_\mathrm{e}}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_{N_\mathrm{e}}^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_1}^{(1)},\ \mathbf{E}_{N_\mathrm{e}}^{\mathrm{no}}\boldsymbol{\zeta}_{u,2}\right) \\
\dots & \dots & \dots \\
\mathcal{R}_1^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_j}^{(k)},\ \mathbf{E}_1^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) & \dots & \mathcal{R}_{N_\mathrm{e}}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_{N_\mathrm{e}}^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_j}^{(k)},\ \mathbf{E}_{N_\mathrm{e}}^{\mathrm{no}}\boldsymbol{\zeta}_{u,n}\right) \\
\dots & \dots & \dots \\
\dots & \dots & \dots \\
\mathcal{R}_1^{\sigma,\mathrm{hf}}\left(\mathbf{E}_q^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_{n_\mathrm{train}}}^{(K)},\ \mathbf{E}_1^{\mathrm{no}}\boldsymbol{\zeta}_{u,N_u}\right) & \dots & \mathcal{R}_{N_\mathrm{e}}^{\sigma,\mathrm{hf}}\left(\mathbf{E}_{N_\mathrm{e}}^{\mathrm{qd}}\boldsymbol{\sigma}_{\mu_{n_\mathrm{train}}}^{(K)},\ \mathbf{E}_{N_\mathrm{e}}^{\mathrm{no}}\boldsymbol{\zeta}_{u,N_u}\right)
\end{array}\right]
$$

Appendix B

Derivation and computation on error indicator for ROMs in structural mechanics

Within this appendix, we delve into a more intricate exploration of the explicit computation of error indicators. Section B.1 specifically addresses the expansion of error indicators to encompass time-dependent scenarios. We thoroughly introduce the error indicators practically employed to steer the greedy process. Additionally, in Section B.2, we put forth a proposal for extending the error indicator to the multi-modeling scenario. It's important to note that this extension has not undergone numerical validation and demands further research for thorough verification.

B.1 Computation of error indicator for a time-dependent single-modeling case

B.1.1 Time-dependent external forces

In the following section, we consider a formulation where the external loading can vary during time. We describe the modification of the methodology for computing the designed error indicator. In such a situation, we have a different linear form for each timestep. We can then define:

$$\left(\psi^{\sigma,(k)}_{N_\sigma+1},\ v\right) = \mathcal{L}_{N_\sigma+1}(v), \quad \forall v \in \mathcal{X}^{\mathrm{hf}}_{\mathrm{bc}} \quad \text{with} \quad \mathcal{L}^{(k)}_{N_\sigma+1} = \int_{\Omega} f^{(k)}_v \cdot v \; dx + \int_{\Gamma_n} f^{(k)}_s \cdot v \; ds$$

This leads to the modification of the Gramian matrix for the last column and the last row:

$$\begin{cases} \forall n, m \in \{1, ..., N_\sigma\}, & \left(\boldsymbol{\Sigma}^{(k)}_N\right)_{n,m} = \left(\boldsymbol{\Sigma}_N\right)_{n,m} \\ \forall n \in \{1, ..., N_\sigma + 1\}, & \left(\boldsymbol{\Sigma}^{(k)}_N\right)_{n,N_\sigma+1} = \left(\boldsymbol{\Sigma}^{(k)}_N\right)_{N_\sigma+1,n} = \left(\psi^{\sigma}_n,\ \psi^{\sigma,(k)}_{N_\sigma+1}\right) \end{cases}$$

In practice, we can observe that the $N_\sigma \times N_\sigma$ upper-left submatrix doesn't change over time. A cost-efficient implementation of the Gramian matrix would be only to change the appropriate row over time.

B.1.2 Normalisation of the error indicator

In order not to have values of dual norms that differ depending on the order of magnitude of the loading, we choose to normalize the residual using the norm of the Riesz elements for the external loadings. Moreover, this choice seems consistent with the relative convergence criteria used in pratice in `code_aster` (see Appendix A.1.3). We define $\widetilde{\mathbf{\Sigma}}_N \in \mathbb{R}^{N_\sigma+1, N_\sigma+1}$:

$$\forall n, m \in \{1, ..., N_\sigma\}, \quad \left(\widetilde{\mathbf{\Sigma}}_N^{(k)}\right)_{n,m} = \frac{\left(\mathbf{\Sigma}_N^{(k)}\right)_{n,m}}{\left(\mathbf{\Sigma}_N^{(k)}\right)_{N_\sigma+1, N_\sigma+1}} = \frac{\left(\mathbf{\Sigma}_N^{(k)}\right)_{n,m}}{\left\|\psi_{N_\sigma+1}^{\sigma,(k)}\right\|^2}$$

The actual error indicator used in our computations is:

$$\Delta_{N,\mu}^{\text{av}} = \sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(\Delta_{N,\mu}^{(k)}\right)^2}, \quad \text{with} \quad \left(\Delta_{N,\mu}^{(k)}\right)^2 = \left(\widetilde{\alpha}_{\sigma,\mu}^{(k)}\right)^T \cdot \widetilde{\mathbf{\Sigma}}_N^{(k)} \cdot \widetilde{\alpha}_{\sigma,\mu}^{(k)} \tag{B.1}$$

B.2 Extension of the error indicator for a time-dependent multi-modeling case

The formulation of the error indicator given in the core of the manuscript is designed for a three-dimensional problem. In this section, we offer an extension of the method to the case of a coupled approach. As a reminder, in this case we have two residuals, one for the one-dimensional contribution and one for the three-dimensional contribution. We have two different operators for the 3d and the 1d domains:

$$\begin{aligned}
\mathcal{R}_\mu^\sigma\left(\sigma_\mu^{(k)},\ v\right) &= \int_\Omega \sigma_\mu^{(k)} : \varepsilon\left(v\right)\ d\Omega - \int_\Omega f_v \cdot v\ d\Omega - \int_\Gamma f_s \cdot v\ d\Gamma \\
\mathcal{R}_\mu^N\left(N_\mu^{(k)},\ w\right) &= \int_\mathcal{C} N_\mu^{(k)} : \partial_s w\ ds - \int_\mathcal{C} f_v \cdot w\ ds \underbrace{- \sum_{i=1}^{\mathcal{N}^{1d}}\left(\frac{t^{(k)} - t^{\text{init,p}}}{t^{\text{end,p}} - t^{\text{init,p}}} F\right) \cdot w_i}_{\text{only during a prestressing step}}
\end{aligned}$$

One possible approach could be to reformulate the problem to have a single uniquely defined operator that accounts for multi-modeling. This involves defining a residual form over the entire domain, as follows:

$$\mathcal{R}_\mu^{[\sigma,N]}\left(\begin{bmatrix}\sigma\\N\end{bmatrix}, \begin{bmatrix}v\\w\end{bmatrix}\right) = \mathcal{R}_{\text{int},\mu}^{[\sigma,N]}\left(\begin{bmatrix}\sigma\\N\end{bmatrix}, \begin{bmatrix}v\\w\end{bmatrix}\right) - \mathcal{R}_{\text{ext},\mu}^{[\sigma,N]}\left(\begin{bmatrix}v\\w\end{bmatrix}\right) \tag{B.2}$$

This way, the problem can be seen through the lens of a single operator. The decomposition into generalized force modes constructed from the stresses and normal efforts at each point of the problem then allows for an approach to build an error indicator in the same manner as previously provided. We can indeed have a linear decomposition over the modes in generalized forces. Specifically, the restricted operators on each sub-domain (3d and 1d) are linear with respect to the constraint or normal effort. We encounter the same property that allowed us to derive the error indicator in the three-dimensional case.

Appendix C

Example of Trust-Region approach with quadratic approximation: Application to Rosenbrock function

This appendix reproduces an approach and an example provided in pre-existing work [Zah16] with an implementation whose blocks are reused for the elements presented in this thesis. The interest is to present keys to understanding the trust-region method with the use of quadratic approximation, which is its natural extension.

C.1 Trust-region approach with quadratic approximation

C.1.1 Application to Rosenbrock function

In order to highlight the methodology on a simple case, we adopt the classic illustration case for optimization algorithm demonstrations: the Rosenbrock function. This function is often used to test optimization algorithms. Indeed, this non-linear function has a global minimum $\mu^{\mathrm{f}} = (1, 1)$ inside a parabolic "valley" ("banana"). Figure C.1 depicts this function around its global minimum. Conventional gradient algorithms can have difficulty converging quickly, which motivates tests on this particular function.

Problem formulation

We consider the classic Rosenbrock problem:

$$\forall \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} \in \mathbb{R}^2, \quad F(\mu) = 100\left(\mu_2 - \mu_1^2\right)^2 + (1 - \mu_1)^2 \tag{C.1}$$

The gradient and hessian of $F(\mu)$ are expressed as follows:

$$\forall \mu \in \mathbb{R}^2 \quad \nabla F(\mu) = \begin{bmatrix} -400\mu_1\left(\mu_2 - \mu_1^2\right) - 2\left(1 - \mu_1\right) \\ 200\left(\mu_2 - \mu_1^2\right) \end{bmatrix} \quad \text{et} \quad \nabla^2 F(\mu) = \begin{bmatrix} 400\left(3\mu_1^2 - \mu_2\right) + 2 & -400\mu_1 \\ -400\mu_1 & 200 \end{bmatrix} \tag{C.2}$$

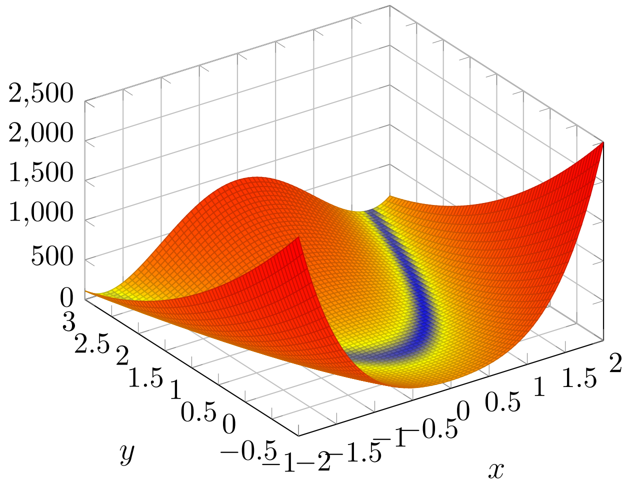


Figure C.1: Visualization of the Rosenbrock function for $(x, y) \in [-2, 2] \times [-1, 3]$

Approximation model

The approximation model function is considered to be a quadratic approximation of $F(\mu)$ with controllable errors introduced in the value and gradient at the expansion point, $\forall \mu \in \mathbb{R}^2$:

$$G\left(\mu, \mu_k, \varepsilon_k, \delta_k\right) = F\left(\mu_k\right) + \varepsilon_k + \left(\nabla F\left(\mu_k\right) + \delta_k 1\right)^T \left(\mu - \mu_k\right) + \left(\mu - \mu_k\right)^T \nabla^2 F\left(\mu_k\right)\left(\mu - \mu_k\right) \quad \text{(C.3)}$$

We rely on the analytical expressions for the gradient and hessian obtained in Equation (C.3).With such a formulation, the parameter ε_k (resp. δ_k) may be interpreted as the error when evaluating the function (resp. function gradient) at the center of the trust region:

$$G\left(\mu_k, \mu_k, \varepsilon_k, \delta_k\right) = F\left(\mu_k\right) + \varepsilon_k, \qquad \nabla G\left(\mu_k, \mu_k, \varepsilon_k, \delta_k\right) = \nabla F\left(\mu_k\right) + \delta_k 1 \quad \text{(C.4)}$$

The approximation model $m_k(\mu)$ is considered to be a quadratic (yet inexact) approximation of $m_k(\mu)$ centered within the confidence region. The constraint for the trust-region ($\vartheta_k(\mu)$) is established relying on the exact point error associated with the objective function, while the gradient error indicator ($\varphi_k(\mu)$) is derived from the exact gradient error:

$$\forall \mu \in \mathbb{R}^2, \qquad \left\{ \begin{array}{rcl} m_k\left(\mu\right) & = & G\left(\mu, \mu_k, \varepsilon_k, \delta_k\right) \\ \vartheta_k\left(\mu\right) & = & \left|F\left(\mu\right) - m_k\left(\mu\right)\right| + \left|F\left(\mu_k\right) - m_k\left(\mu_k\right)\right| \\ \varphi_k\left(\mu\right) & = & \left\|\nabla F\left(\mu\right) - \nabla m_k\left(\mu\right)\right\| \end{array} \right. \quad \text{(C.5)}$$

The gradient of the approximation model is defined as such:

$$\nabla m_k\left(\mu\right) = \nabla F\left(\mu_k\right) + \delta_k + \nabla^2 F\left(\mu_k\right)\left(\mu - \mu_k\right) \quad \text{(C.6)}$$

The expression for the trust region constraint and the gradient error indicator at the center of the trust region are :

$$\left\{ \begin{array}{rcl} \vartheta_k\left(\mu_k\right) & = & 2\varepsilon_k \\ \varphi_k\left(\mu_k\right) & = & \sqrt{2}\delta_k \end{array} \right. \quad \text{(C.7)}$$

The expression of the trust region constraint at the center of the trust region is :

$$\begin{aligned}\vartheta_k\left(\mu_k\right) &= \left|F\left(\mu_k\right)-m_k\left(\mu_k\right)\right|-\left|F\left(\mu_k\right)-m_k\left(\mu_k\right)\right| \\ &= 2\left|F\left(\mu_k\right)-m_k\left(\mu_k\right)\right| \\ &= 2\varepsilon_k\end{aligned}$$

awhile the expression for the gradient error indicator at the center of the trust region is :

$$\begin{aligned}\varphi_k\left(\mu_k\right) &= \left\|\nabla F\left(\mu\right)-\nabla m_k\left(\mu_k\right)\right\| \\ &= \left\|\nabla F\left(\mu\right)-\nabla F\left(\mu_k\right)-\delta_k 1\right\| \\ &= \sqrt{2}\delta_k\end{aligned}$$

In order to respect the constraints for global convergence and knowing the expression given by Equation (C.5), we have the constraints on choice ε_k and δ_k:

$$\left\{\begin{array}{lll}\varepsilon_k & \leq & \dfrac{\kappa_\vartheta}{2}\Delta_k \\ \\ \delta_k & \leq & \dfrac{\kappa_\varphi}{\sqrt{2}}\min\left\{\left\|\nabla m_k\left(\mu_k\right)\right\|,\ \Delta_k\right\}\end{array}\right. \tag{C.8}$$

$m_k\left(\mu\right)$	$G\left(\mu,\mu_k,\varepsilon_k,\delta_k\right)$
$\vartheta_k\left(\mu\right)$	$\left\|F\left(\mu\right)-m_k\left(\mu\right)\right\|+\varepsilon_k$
ε_k	$\dfrac{\kappa_\vartheta}{2}\Delta_k$
δ_k	Backtracking Linesearch sur : $\dfrac{\kappa_\varphi}{\sqrt{2}}\min\left\{\left\|\nabla m_k\left(\mu_k\right)\right\|,\ \Delta_k\right\}$

Table C.1: Summary of approximation model choices at each iteration.

Solver for the trust-region subproblem

When using the BFGS interior point method with linear search, we must derive the gradient of the logarithmic barrier function associated with the optimization problem. The gradient is determined as follows:

$$\begin{aligned}\nabla\phi_k^\gamma\left(\mu\right) &= \nabla m_k\left(\mu\right)-\gamma\nabla\left(\log\left[\Delta_k-\vartheta_k\left(\mu_k\right)\right]\right) \\ &= \nabla m_k\left(\mu\right)-\gamma\left(\frac{1}{\Delta_k-\vartheta_k\left(\mu\right)}\nabla\left(\Delta_k-\vartheta_k\left(\mu\right)\right)\right) \\ &= \nabla m_k\left(\mu\right)+\frac{\gamma}{\Delta_k-\vartheta_k\left(\mu\right)}\nabla\vartheta_k\left(\mu\right)\end{aligned}$$

We need to have en expression of the gradient $(F(\mu) - m_k(\mu))$ for our specific choide of the approximation model (Equation (C.5)):

$$\begin{aligned}\nabla\left(F(\mu) - m_k(\mu)\right) &= \nabla F(\mu) - \nabla m_k(\mu) \\ &= \nabla F(\mu) - \left[\nabla F(\mu_k) + \delta_k + \nabla^2 F(\mu_k)(\mu - \mu_k)\right] \\ &= \nabla F(\mu) - \nabla F(\mu_k) - \delta_k - \nabla^2 F(\mu_k)(\mu - \mu_k).\end{aligned}$$

Thus, the gradient of the trust-region constraint can be written:

$$\begin{aligned}\nabla\vartheta_k(\mu) &= \nabla\left(|F(\mu) - m_k(\mu)| + \varepsilon_k\right) \\ &= \nabla\left(|F(\mu) - m_k(\mu)|\right) \\ &= \operatorname{sgn}\left(F(\mu) - m_k(\mu)\right)\left[\nabla F(\mu) - \nabla F(\mu_k) - \delta_k - \nabla^2 F(\mu_k)(\mu - \mu_k)\right]\end{aligned}$$

$\nabla\phi_k^\gamma(\mu)$	$-\nabla m_k(\mu) + \frac{\gamma}{\Delta_k - \vartheta_k(\mu_k)}\nabla\vartheta_k(\mu_k)$
$\nabla\vartheta_k(\mu_k)$	$\operatorname{sgn}\left(F(\mu) - m_k(\mu)\right)\left[\nabla F(\mu) - \nabla F(\mu_k) - \delta_k - \nabla^2 F(\mu_k)(\mu - \mu_k)\right]$

Table C.2: Summary of the gradient expression needed to solve the problem numerically in the trust-region.

Example of a parameter set

In this section, we present some numerical results regarding the minimization of the Rosenbrock function. Table C.2 summarizes the parameters employed in the optimization algorithm for minimizing the Rosenbrock function. We opt to categorize the parameters into two parts: those associated with the trust region method and others specific to the numerical resolution method BFGS. It is important to note that we did not aim to obtain an optimal set of parameters. Indeed, the purpose of this document is solely to gain insight into the algorithm's behavior. Nevertheless, we acknowledge that the convergence speed is highly dependent on the hyperparameters.

κ_ϑ	κ_φ	η_1	η_2	γ	Δ_0	$\Delta_{\max}$
0.5	1.0	0.3	0.6	0.5	1	$1e8$

(a) Parameters for the trust-region.

c	τ	γ^{r}
10^{-4}	10^{-1}	10^{-5}

(b) Parameters for the BFGS algorithm.

Figure C.2: Parameters used for optimization of Rosenbrock functions.

Figure C.3 highlights the successive centers obtained when using the trust region method. The initial point provided for our test case is $(0, 1)$, while the final value is $(1, 1)$. Figure C.4 shows the convergence of the center values towards the final value. Table C.2 summarizes the exact outputs of the different iterations (center of the confidence region, candidate provided by the sub-problem solution and associated Boolean acceptance of the candidate). It is clear that the algorithm will steer into the "valley" and gradually work its way up to the optimum. This convergence monitoring is quite satisfactory.

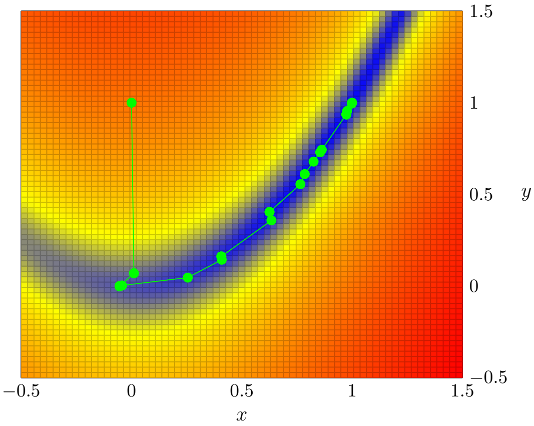


Figure C.3: Visualization of trust region centers over iterations from an initial point $\mu_0 = (0, 1)$ to the global minimum $\mu^{\mathrm{f}} = (1, 1)$

Figure C.5 represents the values of the functionals (objective function and approximation model) and gradients (objective function and approximation model) at the centers of the trust region. We notice that the values of the norms of the gradient (Figure C.5a) and the functional (Figure C.5b) both tend towards 0 (the global minimum of the function is zero, while the local minimum convergence implies a cancellation of the gradient).

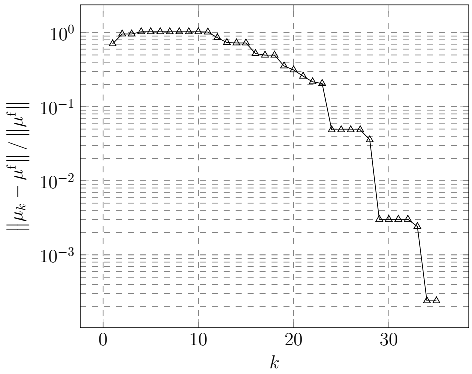
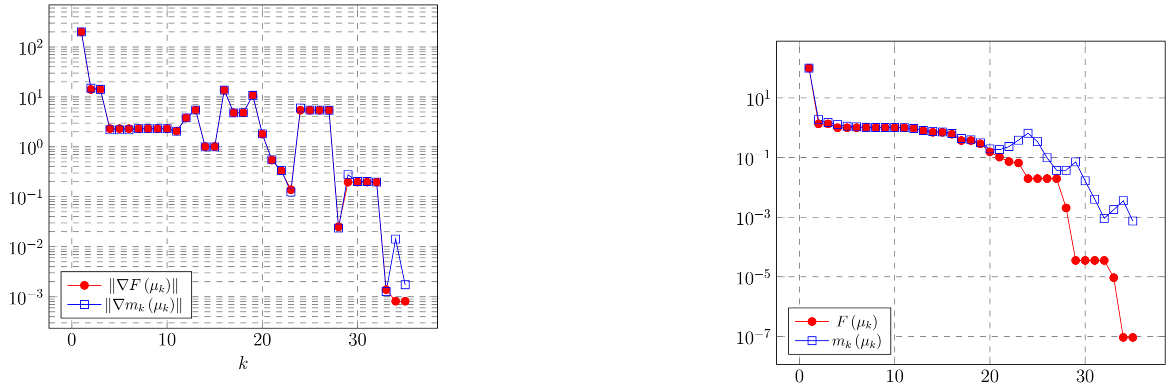


Figure C.4: Evolution of the relative error between the center of the confidence region at iteration k and the optimal value.

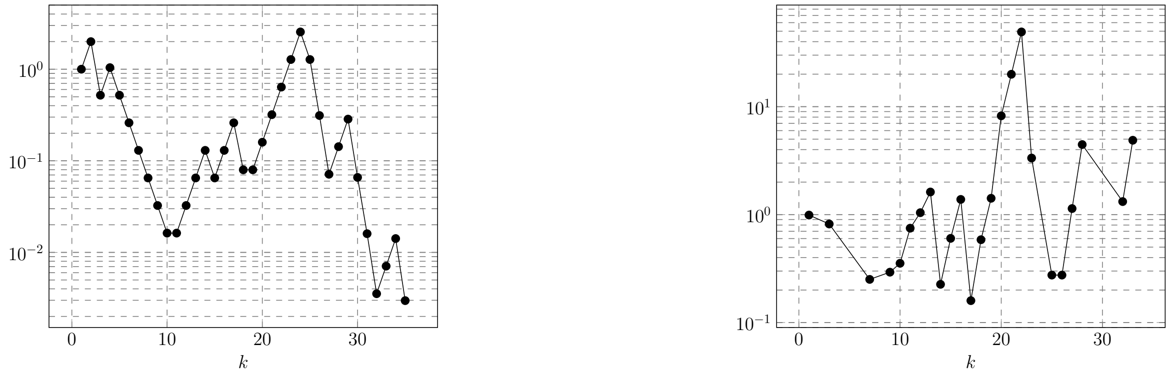
Figure C.6 depicts certain parameters specific to the trust region method throughout the iterations. Figure C.6a illustrates the radii of the trust regions, while Figure C.6 corresponds to the ratio used for updating the trust region. The variations in these parameters showcase the algorithm's behavior over the course of iterations. The trust region expands when there is a sequence of highly successful steps (evidenced by high values of the ratio ρ_k around the twentieth iteration) and contracts when the outcome is less favorable. Monitoring these parameters is crucial for verifying the correct implementation and ensuring a judicious choice of hyperparameters.



(a) Evolution of gradients over iterations for the objective function and the approximation model.

(b) Model evolution over iterations for objective function and approximation model.

Figure C.5: Evolution of model evaluations and gradients over iterations for the objective function F and the approximation model m_k at the centers of the trust-region μ_k.



(a) Radius of the trust-region Δ_k

(b) Update ratio ρ_k

Figure C.6: Evolution of the radius of the Δ_k trust region and ratio for ρ_k update over the iterations.

Iteration	$(\mu_k)_1$	$(\mu_k)_2$	$(\mu_k^*)_2$	$(\mu_k^*)_2$	**Update**
1	0.01071657	0.06975253	0.01071657	0.06975253	True
2	0.01071657	0.06975253	0.16655578	-0.03717835	False
3	-0.05429478	-0.00082821	-0.05429478	-0.00082821	True
4	-0.05429478	-0.00082821	0.28009449	-0.00254455	False
5	-0.05429478	-0.00082821	0.2303478	0.00607251	False
6	-0.05429478	-0.00082821	0.1839345	0.0133266	False
7	-0.05429478	-0.00082821	0.12799273	0.02212548	False
8	-0.05429478	-0.00082821	0.06844636	0.03123113	False
9	-0.05429478	-0.00082821	-0.03037037	0.00714035	False
10	-0.04279195	0.00297439	-0.04279195	0.00297439	True
11	0.25475199	0.04597492	0.25475199	0.04597492	True
12	0.40887598	0.14338131	0.40887598	0.14338131	True
13	0.40725865	0.16131494	0.40725865	0.16131494	True
14	0.40725865	0.16131494	0.65105892	0.37165572	False
15	0.63422116	0.35806608	0.63422116	0.35806608	True
16	0.62543794	0.40429111	0.62543794	0.40429111	True
17	0.62543794	0.40429111	0.82832825	0.6400577	False
18	0.76546561	0.55543945	0.76546561	0.55543945	True
19	0.78511275	0.61062513	0.78511275	0.61062513	True
20	0.82519653	0.67884287	0.82519653	0.67884287	True
21	0.85494312	0.72954289	0.85494312	0.72954289	True
22	0.86243455	0.74323706	0.86243455	0.74323706	True
23	0.97396504	0.93611906	0.97396504	0.93611906	True
24	0.97396504	0.93611906	0.62559565	0.27083389	False
25	0.97396504	0.9361190	0.9392685	0.8805215	False
26	0.97396504	0.93611906	0.93926951	0.88052354	False
27	0.97719616	0.9548519	0.97719616	0.9548519	True
28	0.99826813	0.9960903	0.99826813	0.9960903	True
29	0.99826813	0.9960903	0.86242389	0.72482268	False
30	0.99826813	0.9960903	0.98610839	0.97221267	False
31	0.99826813	0.9960903	0.98605804	0.97211151	False
32	0.99847248	0.99694064	0.99847248	0.99694064	True
33	0.99984926	0.99969614	0.99984926	0.99969614	True
34	0.99984926	0.99969614	0.98500128	0.96995621	False
35	0.99984926	0.99969614	0.99850938	0.99701424	False

Table C.3: Iteration table for minimizing the Rosenbrock function.

Appendix **D**

Details for PDE-constrained optimization algorithms

D.1 Derivation of the adjoint equation for PDE-constrained optimization

As a reminder, we focus on the case of an HF cost function to present the derivation of an adjoint equation. We are interested in the PDE-constrained optimization problem:

$$\begin{aligned} \min_{\mu \in \mathbb{R}^p} \quad & J^{\mathrm{hf}}\left(\mu\right) = F\left(\mathbf{u}_\mu,\ \mu\right) \\ \text{subject to} \quad & \mathbf{R}^{\mathrm{hf}}\left(\mathbf{u}_\mu,\ \mu\right) = 0 \end{aligned}$$

We refer to the works [Joh12] or [Zah16] for more details. A first method is to derive an adjoint approach using an algebraic trick. We can write:

$$\begin{aligned} \frac{dJ^{\mathrm{hf}}}{d\mu} &= \frac{\partial F}{\partial \mu} - \frac{\partial F}{\partial \mathbf{u}} \left[\frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\right]^{-1} \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu} \\ &= \frac{\partial F}{\partial \mu} - \left[\left[\frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\right]^{-\top} \left[\frac{\partial F}{\partial \mathbf{u}}\right]^{\top}\right]^{\top} \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu} \\ &= \frac{\partial F}{\partial \mu} - \mathbf{\Lambda}_\mu^{\top} \frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mu}. \end{aligned}$$

This Lagrange multiplier is derived from the solution of an equation, called the adjoint equation, where we seek to determine the solution to the following problem:

$$\left[\frac{\partial \mathbf{R}^{\mathrm{hf}}}{\partial \mathbf{u}}\left(\mathbf{u}_\mu, \mu\right)\right]^{\top} \mathbf{\Lambda}_\mu = \left[\frac{\partial F}{\partial \mathbf{u}}\left(\mathbf{u}_\mu, \mu\right)\right]^{\top} \tag{D.1}$$

Other approaches explain in more detail the origin of this algebraic trick and equation. We briefly present the derivation from sensitivity analysis. This method is also known as the direct method. This method relies on the appearance of a quantity called sensitivity, which corresponds to the derivative of the solution as a function of the parameter. However, in practice, this quantity is rather difficult, if not impossible, to determine. Applying the chain rule to the cost functional J^{hf}, we have:

$$\frac{dJ^{\text{hf}}}{d\mu}(\mu) = \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) + \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\frac{\partial \mathbf{u}}{\partial \mu}(\mu). \tag{D.2}$$

As a reminder, the displacement solution checks the PDE:

$$\mathbf{R}^{\text{hf}}(\mathbf{u}_\mu, \mu) = \mathbf{0}, \quad \forall \mu \in \mathcal{P},$$

Then, we can perform the derivation with respect to the parameter vector:

$$\frac{d\mathbf{R}^{\text{hf}}}{d\mu}(\mathbf{u}_\mu, \mu) = 0 \quad \Rightarrow \quad \frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) + \frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\frac{\partial \mathbf{u}}{\partial \mu}(\mu) = 0, \tag{D.3}$$

- First derivation: Since the Jacobian of the problem is invertible, we can express the following sensitivities:

  $$\frac{\partial \mathbf{u}}{\partial \mu}(\mu) = -\left[\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]^{-1}\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) \tag{D.4}$$

  Therefore, we can combine the equation for the sensitivity and the equation for the gradient of J^{hf}. Such a combination can lead to an expression of the gradient of the cost-function:

  $$\frac{dJ^{\text{hf}}}{d\mu}(\mu) = \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) - \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\left[\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]^{-1}\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) \tag{D.5}$$

  If we define the adjoint variable $\mathbf{\Lambda}$ such that:

  $$\mathbf{\Lambda}^\top = -\frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\left[\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]^{-1} \tag{D.6}$$

  we can thus define the gradient:

  $$\frac{dJ^{\text{hf}}}{d\mu}(\mu) = \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) + \mathbf{\Lambda}^\top\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu)$$

- Second derivation: we introduce a test function $\mathbf{\Lambda}$, and we can multiply the sensitivity equation (Equation (D.3)), which gives:

  $$\mathbf{\Lambda}^\top\left[\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) + \frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\frac{\partial \mathbf{u}}{\partial \mu}(\mu)\right] = 0$$

  We can add these zero terms to the gradient expression Equation (D.2), this gives us the following expression:

$$\begin{aligned}\frac{dJ^{\text{hf}}}{d\mu}(\mu) &= \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) + \frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\frac{\partial \mathbf{u}}{\partial \mu}(\mu)\underbrace{-\mathbf{\Lambda}^\top\left[\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) + \frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\frac{\partial \mathbf{u}}{\partial \mu}(\mu)\right]}_{=0}, \\ &= \frac{\partial F}{\partial \mu}(\mathbf{u}_\mu, \mu) - \mathbf{\Lambda}^\top\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mu}(\mathbf{u}_\mu, \mu) + \underbrace{\left[\frac{\partial F}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu) - \mathbf{\Lambda}^\top\frac{\partial \mathbf{R}^{\text{hf}}}{\partial \mathbf{u}}(\mathbf{u}_\mu, \mu)\right]}_{=0,\ \text{if } \mathbf{\Lambda} \text{ is solution of adjoint equation}}\frac{\partial \mathbf{u}}{\partial \mu}(\mu)\end{aligned}$$

In the case where the multiplier meets the adjoint equation (D.1), we have an expression for the gradient of the cost function.

Appendix E

Additional numerical analysis on trust-region approches relying on ROMs

E.1 Additional numerical investigations for the data-assimilatio problem in linear elasticity

E.1.1 High-fidelity data-assimilation problem: holed-plate under traction

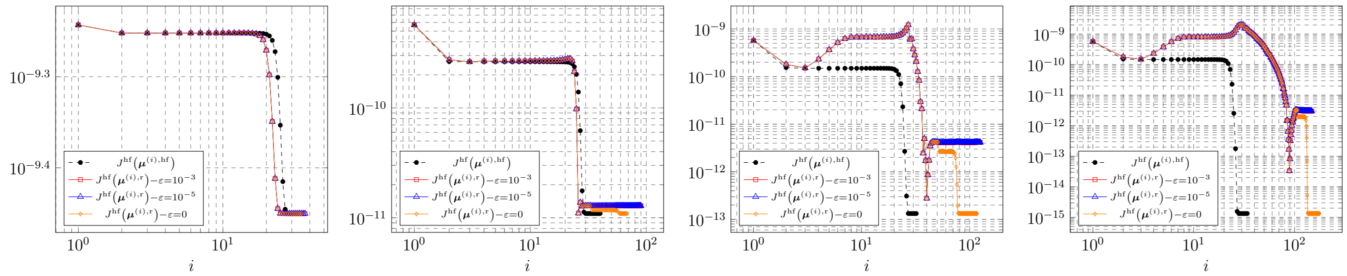
ϕ	$\overline{\sigma}_{\text{err},\mu}$	E_b	ν_b	E_a	ν_a	$J^{\text{hf}}(\mu_a)$	CPU Time (s)	n_{it}	n_{eval}
1	2%	$51 \cdot 10^3$	0.306	$50.5581827 \cdot 10^3$	0.305484477	$3.541475508817821 \cdot 10^{-10}$	13.941	33	79
10	2%	$51 \cdot 10^3$	0.306	$49.9461768 \cdot 10^3$	0.302487792	$1.0934289386267248 \cdot 10^{-11}$	14.931	40	82
10^2	2%	$51 \cdot 10^3$	0.306	$49.9987253 \cdot 10^3$	0.300050883	$1.3300032570281602 \cdot 10^{-13}$	14.855	32	87
10^3	2%	$51 \cdot 10^3$	0.306	$49.9999871 \cdot 10^3$	0.300000514	$1.3347642303932904 \cdot 10^{-15}$	9.949	34	54
1	10%	$55 \cdot 10^3$	0.33	$52.7726000 \cdot 10^3$	0.327046401	$8.01341300712998 \cdot 10^{-9}$	11.988	32	68
10	10%	$55 \cdot 10^3$	0.33	$49.7485497 \cdot 10^3$	0.311232059	$2.3989749766700176 \cdot 10^{-10}$	10.532	35	58
10^2	10%	$55 \cdot 10^3$	0.33	$49.9945140 \cdot 10^3$	0.300218903	$2.860398658370181 \cdot 10^{-12}$	16.947	38	87
10^3	10%	$55 \cdot 10^3$	0.33	$49.9999445 \cdot 10^3$	0.300002211	$2.869206180587488 \cdot 10^{-14}$	11.537	38	61
1	25%	$62.5 \cdot 10^3$	0.375	$56.8999552 \cdot 10^3$	0.365753798	$4.2284561246367983 \cdot 10^{-8}$	20.097	35	114
10	25%	$62.5 \cdot 10^3$	0.375	$49.4526953 \cdot 10^3$	0.323362662	$1.1980140666316695 \cdot 10^{-9}$	32.235	61	168
10^2	25%	$62.5 \cdot 10^3$	0.375	$49.9893620 \cdot 10^3$	0.300424251	$1.3854000701201777 \cdot 10^{-11}$	22.876	59	114
10^3	25%	$62.5 \cdot 10^3$	0.375	$49.9998926 \cdot 10^3$	0.300004280	$1.3887055982309812 \cdot 10^{-13}$	15.538	41	82

Table E.1: Ouputs of the DA process (for different pairs of hyper-parameters) for a solution using 'L-BFGS-B'.

E.1.2 Data-assimilation with ROM and without trust-regions

On the use of various POD tolerances

We report here the results of DA processes using ROM (without TR) for different values of POD tolerances. We have simulated the different DA problems successively, only adjusting the POD tolerance as a hyper-parameter of the problem. In this way, we can investigate the sensitivity of the solution to this parameter. Figure E.1 depicts several evolutions of the cost functions for different values of POD tolerances (and for different inverse problems, more or less regularised). For the most regularised problems, i.e. those where the roughing term is most significant, we do not have



(a) $\phi = 1$ (b) $\phi = 10$ (c) $\phi = 100$ (d) $\phi = 10^3$

Figure E.1: Visualisation of HF (J^{hf}) and reduced (J_k^{r}) cost functions for different values of ϕ through iterations for $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$. The parameter sets $\{\mu^{(i),\mathrm{r}}\}_i$ (resp. $\{\mu^{(i),\mathrm{hf}}\}_i$) correspond to the trajectory obtained for a reduced calculation (resp. HF).

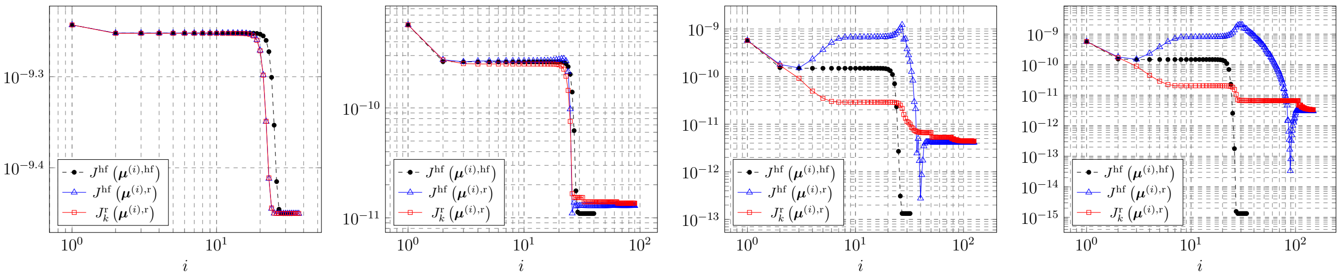
a major differnce on the convergence of the algorithm. The close the problem is to the true inverse problem, the smaller the value of the local minimum and the greater the difference between the solvers' performances. More precisely, we can observe that for a zero POD tolerance, we iteratively enrich the ROM until we reach the convergence threshold obtained for the HF calculation. This is no longer the case for higher POD tolerances. This is consistent since the ROM is thus coarser and cannot reach the same convergence tolerance as the HF model. In the cases observed here, the error on the final solution is fairly small (we can see that the cost functions have extremely low values). Nevertheless, it will ne necessary to study this phenomenon on problems with more complex physics (and more timesteps).

Interest of adaptive enrichment

In this section, we provide an additional analysis of the benefits of iterative basis enrichment in our framework. This complements the analyses provided in Figure 5.9. More specifically, to demonstrate the accuracy of the analysis, we show that this is not dependent on the use of a zero POD tolerance (Figure 5.9) by providing here (Figure E.2) results for a non-zero tolerance. We plot here the convergence curves (on the cost function) for the HF calculation, and for the reduced calculation. For the latter, we provide two plots: fisrt, the evaluation of the HF cost function at iterates in red (which corresponds to what is actually calculated by the minimization algorithm), and second, the evaluation of the HF cost functions at iteates in blue. We can notice that during the first iterations the HF cost functions evaluated at reduced iterates increases from time to time, and so we leave the minimization framework. This is linked to the fact that the reduced model is moving out of its region of validity. However, over the course of the iterations, the models becomes better and the phenomenon fades until convergence is reached.

E.2 Additional results for the data-assimilation problem in hyperelasticity

In the appendix, we provide more insight on an observation made for the four dimensional test case in hyperelasticity. More specifically, we stated that to ease the convergence of the algorithm, we constrained parameter values by providing boundaries. The control of these boundaries is defined by Equation (5.31). In practice, we use slightly more precise constraints to avoid coefficients of

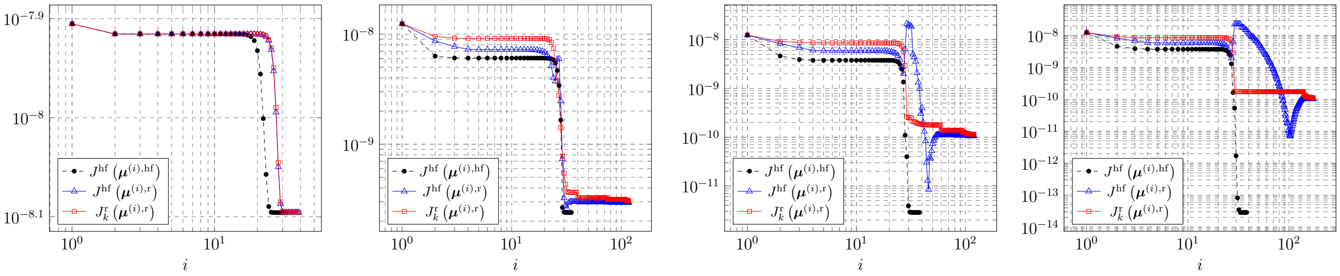


(a) $\phi = 1$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(b) $\phi = 10$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(c) $\phi = 100$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

(d) $\phi = 10^3$ and $\overline{\sigma}_{\mathrm{err},\mu} = 2\%$.

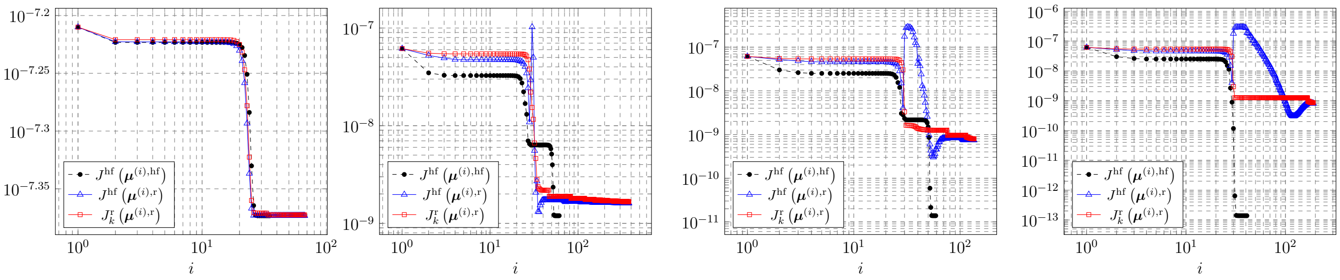


(e) $\phi = 1$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

(f) $\phi = 10$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

(g) $\phi = 100$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.

(h) $\phi = 10^3$ and $\overline{\sigma}_{\mathrm{err},\mu} = 10\%$.



(i) $\phi = 1$ and $\overline{\sigma}_{\mathrm{err},\mu} = 25\%$.

(j) $\phi = 10$ and $\overline{\sigma}_{\mathrm{err},\mu} = 25\%$.

(k) $\phi = 100$ and $\overline{\sigma}_{\mathrm{err},\mu} = 25\%$.

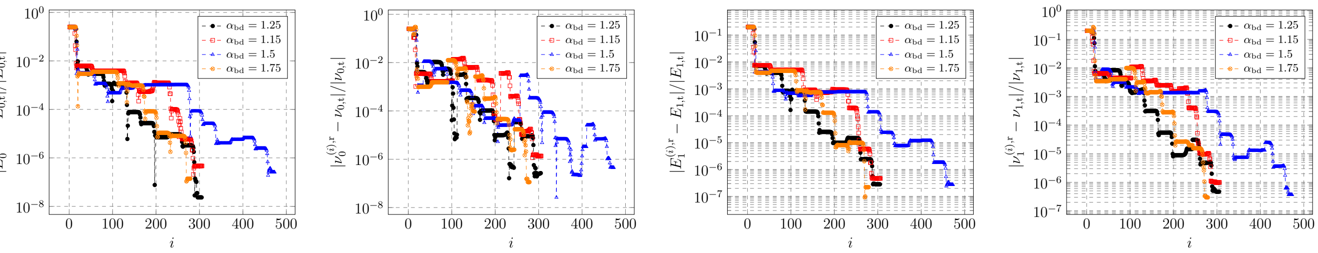
(l) $\phi = 10^3$ and $\overline{\sigma}_{\mathrm{err},\mu} = 25\%$.

Figure E.2: Visualisation of HF (J^{hf}) and reduced (J^{r}_{k}) cost functions for different hyper-parameter pairs (ϕ and $\overline{\sigma}_{\mathrm{err},\mu}$) through iterations. The parameter sets $\{\mu^{(i),\mathrm{r}}\}_i$ (resp. $\{\mu^{(i),\mathrm{hf}}\}_i$) correspond to the trajectory obtained for a reduced calculation (resp. HF). For this simulations, the reduced order basis is generated thanks to a tolerance $\varepsilon_u = 10^{-5}$.

Poisson or Young's modulus taking on non-physical values:

$$\forall i \in \{0,1\}, \quad \left\{ \begin{aligned} E_i^{\max} &= \mu_{\mathrm{b}} + \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}}, \quad \text{and} \quad E_i^{\min} = \max\left(\mu_{\mathrm{b}} - \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}}, 0\right) \\ \nu_i^{\max} &= \min\left(\mu_{\mathrm{b}} + \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}}, 0.5\right), \quad \text{and} \quad \nu_i^{\min} = \max\left(\mu_{\mathrm{b}} - \alpha_{\mathrm{bd}} \frac{\overline{\sigma}_{\mathrm{err},\mu}}{100} \mu_{\mathrm{b}}, -1\right) \end{aligned} \right.$$

Figure E.3 depicts several trajectories of relative errors compared to the expected value for various alpha values. We can observe that an optimal choice of boundaries reduces the number of algorithm iterations and improves the approximation error.



(a) Errors on E_0 iterates.

(b) Errors on ν_0 iterates.

(c) Errors on E_1 iterates.

(d) Errors on ν_1 iterates.

Figure E.3: Evolution of the parameters and the relative errors on the components of the parameter vector comparing to the true parameter for $\Delta_0 = 10^{-1}$ for `'classical'` approach on the overall field for a four dimensional test case for $(\overline{\sigma}_{\mathrm{err},\mu},\ \phi) = (25\%, 1000)$ for different values of α_{bd}. The latter is used to defined the boundaries for the optimization problem.

Appendix F

Alternatives ideas and algorithms for incremental hyper-reduction approaches

In this appendix, we provide additional insights into incremental hyper-reduction approaches to expand upon the topic discussed in Chapter 6. More specifically, we focus on two subjects. First, we present some supplementary numerical results on the purely non-intrusive incremental approach mentioned earlier. This serves to illustrate certain choices made in the methodology. Secondly, we present a theoretical derivation establishing a connection between a low-rank approximation and a NNLS optimization problem. We do not provide numerical results for this case, but rather offer some ideas to pave the way for future considerations on the subject

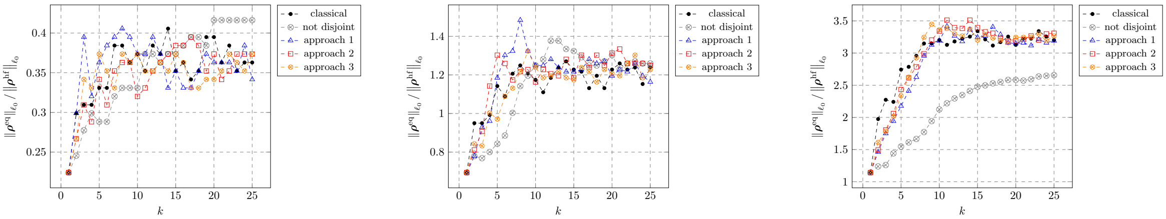
F.1 Numerical experiments on the incremental disjoint approach

In this section, we present additional numerical results to elucidate the underlying reasons for certain algorithmic choices made in Algorithm 7. Specifically, the objective of this section is to showcase some numerical experiments associated with the application of this algorithm and its variants, demonstrating a degree of applicability of the methodology. Before delving into the details, we clarify certain technical aspects. The results we present are as follows. Firstly, we focus on the case referred to as case 2 in Chapter 6. This entails considering a set of modes generated a priori on a training set of 25 points, testing the incremental aspect by successively adding all parameters to the problem. Secondly, in this context, we do not explicitly apply the numerical implementation of the Lawson algorithm. We consider a scenario where the study is conducted on the GGT matrix. In this context, this approach coupled with the incremental aspect has shown improved numerical stability.

F.1.1 On the use of disjoint sets and redefinition of approximation for the *incr-dd* approach

As stated above, in Chapter 6., we delineated a range of incremental problem-solving strategies, each with varying degrees of complexity and effectiveness: one involving disjoint sets and another addressing potential non-disjoint sets, with an emphasis on modifying the second member and subsequent renormalization.

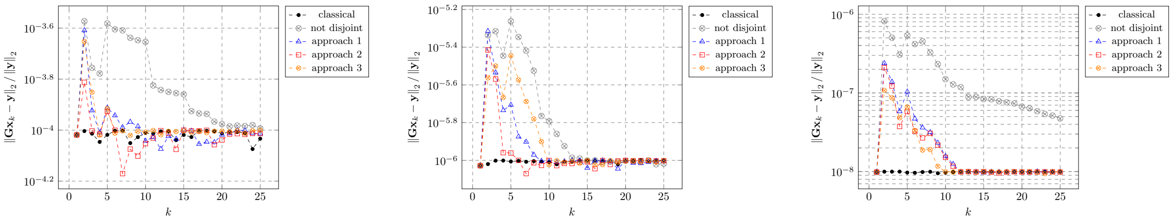
Figure F.1 illustrates the comparative analysis of these diverse methodologies in the case of $n_t =$



(a) Percentage of selected elements for $\delta = 10^{-4}$.

(b) Percentage of selected elements for $\delta = 10^{-6}$.

(c) Percentage of selected elements for $\delta = 10^{-6}$.



(d) Normalized residuals for $\delta = 10^{-4}$.

(e) Normalized residuals for $\delta = 10^{-6}$.

(f) Normalized residuals for $\delta = 10^{-8}$.

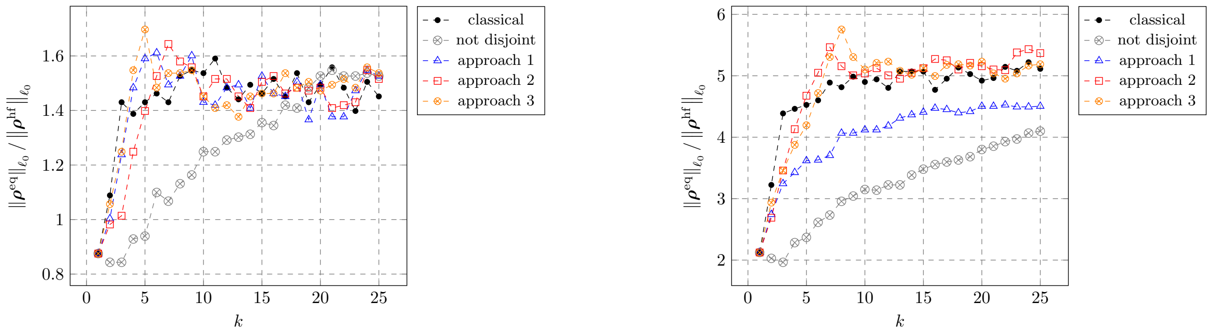
Figure F.1: Comparison of the outputs (percentage of selected elements and normalized residuals) for different approaches for the incr-dd method and comparison with the classical approach for various hyper-reduction tolerance $(n_{\text{train}}, n_t) = (25, 20)$.

20 timesteps and a training set of size $n_{\text{train}} = 25$. Notably, in scenarios involving non-disjoint sets, the incremental process exerts discernible influence on solution quality. Specifically, normalized residuals surpass the prescribed tolerance threshold, leading to a significantly reduced number of selected elements compared to the classical approach (Figures F.1d-F.1f). The introduction of set disjunction ameliorates this discrepancy, yielding a percentage of selected elements that aligns more closely with the reference solution. Irrespective of the chosen approach, it is noteworthy that the use of disjoint sets maintains a solution quality closely approximating that of the classical approach. However, we can acknowledge that initial iterations may yield deviations in normalized errors, which tend to rectify themselves over the course of subsequent iterations.

Furthermore, our desire to establish this observation led us to conduct an experiment involving an augmented number of snapshots linked to each parameter. Specifically, we examined a scenario characterized by a temporal discretization that was doubled ($n_t = 40$). The visual representation in Figure F.2) substantiates the earlier observation regarding the enhancement in approximation quality achieved through the implementation of set disjunction (F.2c-F.2d). Additionally, it is worth noting that, in this context, refraining from updating the second member (referred to as approach 1) resulted in a deterioration of the solution. Consequently, using a disjoint set approach and the subsequent recalculation of the second member in the initial stage of the algorithm (performed during the computation on the updated dictionary) is deemed a suitable course of action.

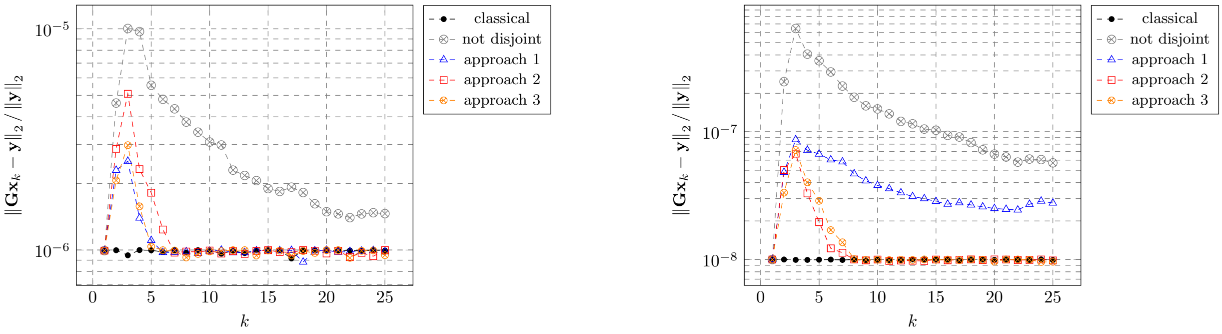
F.1.2 Other solving approaches for sparse representation problem

This approach involves a particular relaxation of the sparse approximation problem, which is much easier to solve in order to reduce the computational cost. Nevertheless, in practice, other numerical approaches may be adopted to solve the approximation problem with a similar idea. Indeed, this kind of problem appears regularly in signal processing and several classes of methods have been



(a) Percentage of selected elements for $\delta = 10^{-6}$.

(b) Percentage of selected elements for $\delta = 10^{-8}$.



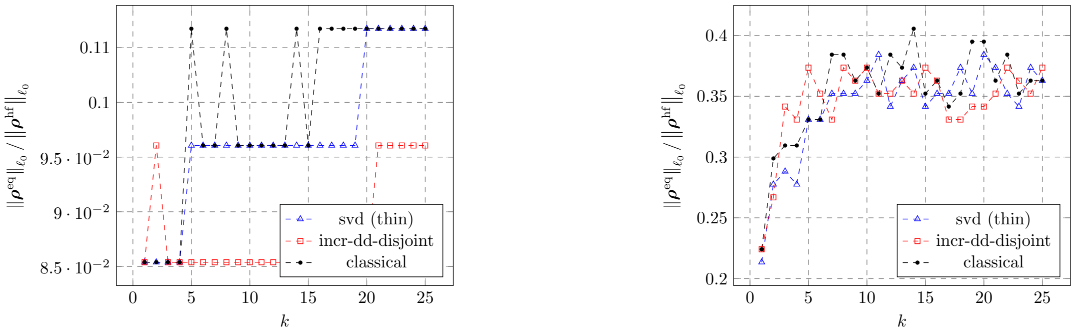
(c) Normalized residuals for $\delta = 10^{-6}$.

(d) Normalized residuals for $\delta = 10^{-8}$.

Figure F.2: Comparison of the outputs (percentage of selected elements and normalized residuals) for different approaches for the incr-dd method and comparison with the classical approach for various hyper-reduction tolerance and for a test-case defined by $(n_{\text{train}}, n_t) = (25, 40)$.

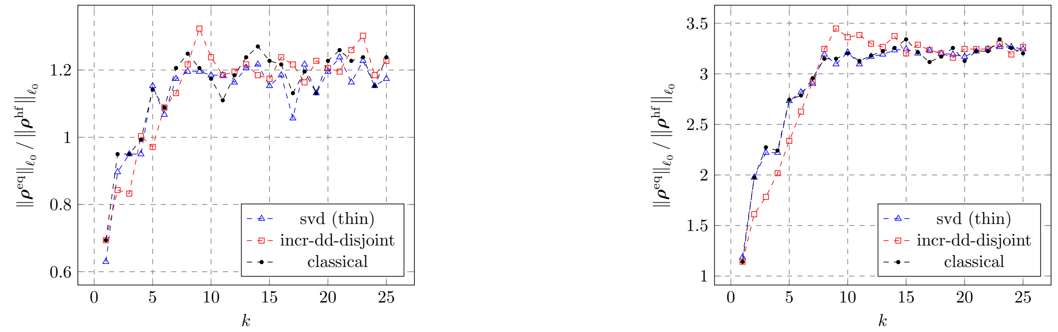
developed to tackle this point. In the literature, approaches such as ℓ_1-norm minimization[YP19], LASSO regularization[CACF17] or Orthogonal Matching Pursuit[CA19] are available to address this kind of mesh sampling problem. By testing our algorithm on at least one alternative algorithm, we open the door to generalizing the method to speed up mesh sampling methodology, even with a different resolution algorithm.

We only present results for the *incr-dd-disjoint* algorithm. Indeed, the low-rank approximation methodology relies on an analysis with estimates in ℓ_2 norm, which does not always stands for alternative algorithms. In our eyes, this algorithm would require some adaptations for other solvers. Moreover, we have shown that the computational speedups are much more interesting for the first algorithm. We present the numerical results for a resolution by Orthogonal Matching Pursuit algorithm[AKSV18][Tro04]. We observe that the speedups remain of good quality for several values of hyper-reduction tolerances (see Figure F.5), wether for the resolution step or the entire procedure.



(a) Percentage of selected elements for $\delta = 10^{-4}$.

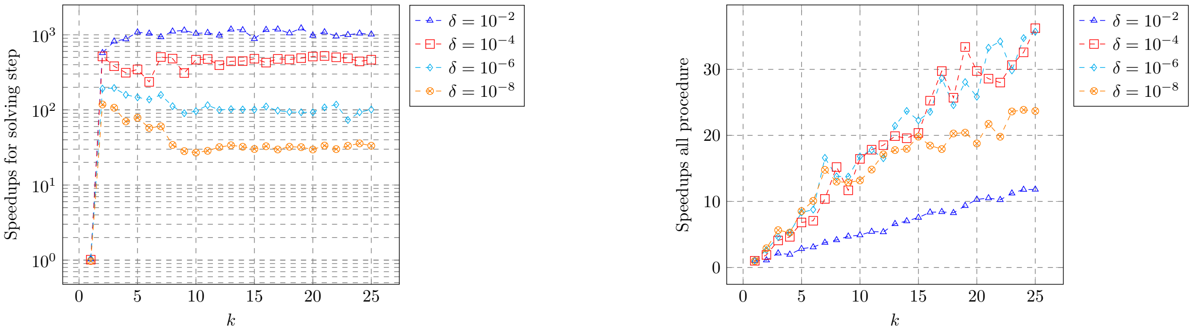
(b) Percentage of selected elements for $\delta = 10^{-6}$.



(c) Percentage of selected elements for $\delta = 10^{-6}$.

(d) Percentage of selected elements for $\delta = 10^{-6}$.

Figure F.3: Comparison of the percentage of selected elements obtained depending on the different method: classical ECSW approach on the full dictionnary (*classical*), use of a thin svd and a low-rank approximation (*svd (thin)*) and use of the incremental dd approach using disjoints sets and a re-normalization (approach denoted by 3 previsouly - *incr-dd-disjoint*).



(a) Incr-dd-disjoint approach.

(b) Incr-dd-disjoint approach.

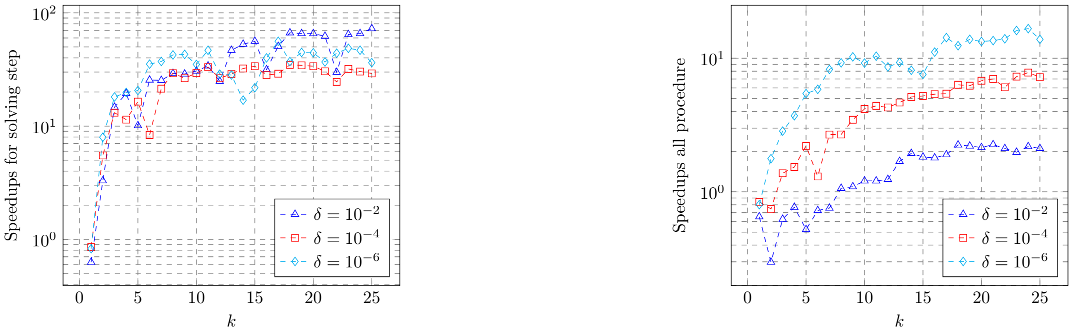
Figure F.4: Speedups for the solving step of the EQ problem using a *incr-dd-disjoint* approach (cf. Figure F.4a) and for the overall procedure (cf. Figure F.4b).

F.2 Low-rank approximation of the dictionnary

F.2.1 Low-rank approximation of G

Use of low-rank and reformulation

Within this section, the objective is to reduce the size of the dictionary to be supplied to the optimization algorithm. From a heuristic standpoint, the information contained in the lines of the



(a) For solving step.

(b) For all procedure.

Figure F.5: Speedups for the solving step of the EQ problem using a *incr-dd-disjoint* approach using an Orthogonal Matching Pursuit Algorithm.

dictionary may be redundant (a line corresponding to a residual for a given snapshot and mode, at a fixed time step). We therefore use a Singular Value Decomposition (SVD) to reduce the dictionary to the minimum amount of information required for the optimization process. Next, we solve an optimization problem for a low-rank approximation of the dictionary containing the signals. We assume that we have an SVD of the dictionary transpose $\mathbf{G}^T \in \mathbb{R}^{N_\text{e}\times n}$:

$$\mathbf{G}^T = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{F.1}$$

where $\mathbf{U} \in \mathbb{R}^{N_\text{e}\times N_\text{e}}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{N_\text{e}\times n}$ and $\mathbf{V}^T \in \mathbb{R}^{n\times n}$. We shall consider this SVD decomposition and assume a truncation at rank k, which means we may rewrite the decomposition in the following matrix form:

$$\mathbf{G}^T = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_1 & 0 \\ 0 & \boldsymbol{\Sigma}_2\end{bmatrix}\begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2\end{bmatrix}^T \tag{F.2}$$

where $\mathbf{U}_1 \in \mathbb{R}^{N_\text{e}\times k}$, $\mathbf{U}_2 \in \mathbb{R}^{m\times(N_\text{e}-k)}$, $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{k\times k}$, $\boldsymbol{\Sigma}_2 \in \mathbb{R}^{(N_\text{e}-k)\times(n-k)}$, $\mathbf{V}_1^T \in \mathbb{R}^{k\times n}$, and $\mathbf{V}_2 \in \mathbb{R}^{(n-k)\times n}$. This decomposition choice matches a truncation in the spectrum, which is a classic method of constructing a low-rank approximation. of a matrix. The orthogonality property of the $\mathbf{V}$ matrix provides the equivalence, for $\boldsymbol{\rho} \in \mathbb{R}^{N_\text{e}}$:

$$\begin{aligned}
\mathbf{G}\boldsymbol{\rho} = \mathbf{y} \quad & \Leftrightarrow \quad \mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^T\boldsymbol{\rho} = \mathbf{y} \\
& \Leftrightarrow \quad \mathbf{V}^T\mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^T\boldsymbol{\rho} = \mathbf{V}^T\mathbf{y} \\
& \Leftrightarrow \quad \left(\boldsymbol{\Sigma}_1\mathbf{U}_1^T + \boldsymbol{\Sigma}_2\mathbf{U}_2^T\right)\boldsymbol{\rho} = \mathbf{V}_1^T\mathbf{y} + \mathbf{V}_2^T\mathbf{y}
\end{aligned}$$

We introduce the following matrices:

$$\mathbf{G}_k = \boldsymbol{\Sigma}_1\mathbf{U}_1^T \quad \text{and} \quad \mathbf{y}_k = \mathbf{V}_1^T\mathbf{y} \tag{F.3}$$

where $\mathbf{G}_k \in \mathbb{R}^{k\times N_\text{e}}$ and $\mathbf{G}_k = \mathbf{V}_1\boldsymbol{\Sigma}_1\mathbf{U}_1^T = \mathbf{V}_1\mathbf{G}_k \in \mathbb{R}^{n\times N_\text{e}}$. With these notations, $\mathbf{G}_k$ defines the k rank approximation of the dictionary. The matrix $\mathbf{G}_k$ is the actual dictionary used for the k rank approximation least squares problem, defined as follows:

$$\boldsymbol{\rho}^\text{eq} = \text{EQ-solve}\left(\mathbf{G}_k,\ \mathbf{y}_k,\ \delta_k\right) \tag{F.4}$$

where δ_k is the tolerance given for this simulated problem. For the moment, we do not specify the rank selection criterion. Assuming we possess a solution to this approximate problem, we need to achieve an approximation on the dictionary. Successive matrix inequalities lead to the following inequality:

$$\left\|\mathbf{G}\boldsymbol{\rho}^{\mathrm{eq}}-\mathbf{y}\right\|_2 \leq \sigma_{k+1}\left(\mathbf{G}\right)\left(\left\|\boldsymbol{\rho}^{\mathrm{eq}}\right\|_2+\left\|\boldsymbol{\rho}^{\mathrm{hf}}\right\|_2\right)+\delta_k\left\|y_k\right\|_2, \tag{F.5}$$

where $\sigma_{k+1}\left(\mathbf{G}\right)$ is the k+1-th singular value of the $\mathbf{G}$ matrix.

F.2.2 Details of the derivation of the low-rank algorithm

In this appendix, we detail the calculations used to derive the various inequalities given in Section F.2.1. As a reminder, $\mathbf{G} \in \mathbb{R}^{n\times N_{\mathrm{e}}}$ is the dictionary for which we are looking for a sparse solution, and $\mathbf{G}_k$ is its low-rank approximation. In the following equations, $\boldsymbol{\rho}^{\mathrm{eq}} \in \mathbb{R}^{N_{\mathrm{e}}}$ is the EQ rule obtained when using the low-rank approximation (see Equation (F.4)). We can derive the inequations:

$$\left\|\mathbf{G}\boldsymbol{\rho}^{\mathrm{eq}}-\mathbf{y}\right\|_2 = \left\|\mathbf{G}\left(\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right)\right\|_2 = \underbrace{\left\|\left(\mathbf{G}-\mathbf{G}_k\right)\left(\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right)\right\|_2}_{\text{(I)}}+\underbrace{\left\|\mathbf{G}_k\left(\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right)\right\|_2}_{\text{(II)}} \tag{F.6}$$

First, we focus on the error analysis of the term (I):

$$\left\|\mathbf{G}_k\left(\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right)\right\|_2 = \left\|\mathbf{G}_k\boldsymbol{\rho}^{\mathrm{eq}}-\mathbf{V}_1\boldsymbol{\Sigma}\mathbf{U}_1^{\top}\boldsymbol{\rho}^{\mathrm{hf}}\right\|_2 = \left\|\mathbf{G}_k\boldsymbol{\rho}^{\mathrm{eq}}-\mathbf{y}_k\right\|_2 \leq \delta_k\left\|\mathbf{y}_k\right\|_2 \tag{F.7}$$

The equality is founded on the following equations:

$$\mathbf{G}_k\boldsymbol{\rho}^{\mathrm{hf}} = \mathbf{V}_1^{\top}\mathbf{G}\boldsymbol{\rho}^{\mathrm{hf}} = \mathbf{V}_1^{\top}\mathbf{y} = \mathbf{y}_k.$$

Second, we focus on the error analysis of the term (II):

$$\left\|\left(\mathbf{G}-\mathbf{G}_k\right)\left(\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right)\right\|_2 \leq \sigma_{k+1}\left(\mathbf{G}\right)\left\|\boldsymbol{\rho}^{\mathrm{eq}}-\boldsymbol{\rho}^{\mathrm{hf}}\right\|_2 \leq \sigma_{k+1}\left(\mathbf{G}\right)\left(\left\|\boldsymbol{\rho}^{\mathrm{eq}}\right\|_2+\left\|\boldsymbol{\rho}^{\mathrm{hf}}\right\|_2\right) \tag{F.8}$$

We can concatenate the two terms of the error bound, leading to the following bound:

$$\left\|\mathbf{G}\boldsymbol{\rho}^{\mathrm{eq}}-\mathbf{y}\right\|_2 \leq \sigma_{k+1}\left(\mathbf{G}\right)\left(\left\|\boldsymbol{\rho}^{\mathrm{eq}}\right\|_2+\left\|\boldsymbol{\rho}^{\mathrm{hf}}\right\|_2\right)+\delta_k\left\|\mathbf{y}_k\right\|_2 \tag{F.9}$$

Bibliography

[AAB$^+$22] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. *arXiv preprint arXiv:2212.14825*, 2022.

[AAB$^+$24a] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, Sylvie Michel-Ponnelle, and Tommaso Taddei. Projection-based model order reduction for prestressed concrete with an application to the standard section of a nuclear containment building. *arXiv preprint arXiv:2401.05098*, 2024.

[AAB$^+$24b] Eki Agouzal, Jean-Philippe Argaud, Michel Bergmann, Guilhem Ferté, and Tommaso Taddei. A projection-based reduced-order model for parametric quasi-static nonlinear mechanics using an open-source industrial code. *International Journal for Numerical Methods in Engineering*, 125(4):e7385, 2024.

[ABH$^+$15] Martin Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E Rognes, and Garth N Wells. The FEniCS Project Version 1.5. *Archive of numerical software*, 3(100), 2015.

[ACMC16] Mehdi Asali, Bruno Capra, Jacky Mazars, and Jean-Baptiste Colliat. Numerical strategy for forecasting the leakage rate of inner containments in double-wall nuclear reactor buildings. *Journal of advanced concrete technology*, 14(8):408–420, 2016.

[ADJLT98] Natalia M Alexandrov, John E Dennis Jr, Robert Michael Lewis, and Virginia Torczon. A trust-region framework for managing the use of approximation models in optimization. *Structural optimization*, 15(1):16–23, 1998.

[AFS00] Eyal Arian, Marco Fahl, and Ekkehard W Sachs. *Trust-region proper orthogonal decomposition for flow control.* Number 2000-2025. Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, 2000.

[AK98] Edoardo Amaldi and Viggo Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1-2):237–260, 1998.

[AKLR18] Harbir Antil, Drew P Kouri, Martin-D Lacasse, and Denis Ridzal. *Frontiers in PDE-constrained Optimization*, volume 163. Springer, 2018.

[AKSV18] Sanjeev Arora, Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. A compressed sensing view of unsupervised text embeddings, Bag-of-n-Grams, and LSTMs. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.

[ALG+01] Natalia M Alexandrov, Robert Michael Lewis, Clyde R Gumbert, Lawrence L Green, and Perry A Newman. Approximation and model management in aerodynamic optimization with variable-fidelity models. *Journal of Aircraft*, 38(6):1093–1101, 2001.

[AT24] Eki Agouzal and Tommaso Taddei. Accelerated construction of projection-based reduced-order models via incremental approaches. *arXiv preprint arXiv:2401.07108*, 2024.

[AZF12] David Amsallem, Matthew J Zahr, and Charbel Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.

[AZW15] David Amsallem, Matthew J Zahr, and Kyle Washabaugh. Fast local reduced basis updates for the efficient reduction of nonlinear systems with hyper-reduction. *Advances in Computational Mathematics*, 41:1187–1230, 2015.

[B+14] Marc Bocquet et al. Introduction to the principles and methods of data assimilation in geosciences. *Notes de cours, École des Ponts ParisTech*, 2014.

[BBB+18] David E-M Bouhjiti, Maxime Boucher, Matthieu Briffaut, Frédéric Dufour, Julien Baroth, and Benoît Masson. Accounting for realistic thermo-hydro-mechanical boundary conditions whilst modeling the ageing of concrete in nuclear containment buildings: Model validation and sensitivity analysis. *Engineering Structures*, 166:314–338, 2018.

[BBI09] Michel Bergmann, Charles-Henri Bruneau, and Angelo Iollo. Enablers for robust POD models. *Journal of Computational Physics*, 228(2):516–538, 2009.

[BC85] Zdenek P Bažant and JC Chern. Concrete creep at variable humidity: constitutive law and mechanism. *Materials and structures*, 18(1):1–20, 1985.

[BC02] François Bouttier and Philippe Courtier. Data assimilation concepts and methods March 1999. *Meteorological training course lecture series. ECMWF*, 718:59, 2002.

[BC08] Michel Bergmann and Laurent Cordier. Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models. *Journal of Computational Physics*, 227(16):7813–7840, 2008.

[BCCF01] Jacques Besson, Georges Cailletaud, Jean-Louis Chaboche, and Samuel Forest. *Mécanique non linéaire des matériaux*. Hermès Science Publ., 2001.

[BCD+11] Peter Binev, Albert Cohen, Wolfgang Dahmen, Ronald DeVore, Guergana Petrova, and Przemyslaw Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. *SIAM journal on mathematical analysis*, 43(3):1457–1472, 2011.

[Ben02] Farid Benboudjema. *Modélisation des déformations différées du béton sous sollicitations biaxiales. Application aux enceintes de confinement de bâtiments réacteurs des centrales nucléaires.* PhD thesis, Université de Marne la Vallée, 2002.

[BFM23] Joshua Barnett, Charbel Farhat, and Yvon Maday. Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility. *Journal of Computational Physics*, 492:112420, 2023.

[BGHvBW03] Lorenz T Biegler, Omar Ghattas, Matthias Heinkenschloss, and Bart van Bloemen Waanders. Large-scale pde-constrained optimization: an introduction. In *Large-Scale PDE-Constrained Optimization*, pages 3–13. Springer, 2003.

[BHL93] Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

[BI19] Michel Bergmann and Angelo Iollo. Sampling and clustering on the POD-Grassmann manifold. In *CSE19-SIAM Conference on Computational Science and Engineering*, 2019.

[Ble18] Jeremy Bleyer. *Numerical Tours of Computational Mechanics with FEniCS*, 2018.

[BLNZ95] Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, 16(5):1190–1208, 1995.

[BMNP04] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T Patera. An 'empirical interpolation'method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[BMP+12] Annalisa Buffa, Yvon Maday, Anthony T Patera, Christophe Prud'homme, and Gabriel Turinici. A priori convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: Mathematical modelling and numerical analysis*, 46(3):595–603, 2012.

[BN72] ZP Bažant and LJ Najjar. Nonlinear water diffusion in nonsaturated concrete. *Matériaux et Construction*, 5:3–20, 1972.

[Bou16] Maxime Boucher. *Analyse de la corrélation spatio-temporelle des déformations entre le coeur d'un ouvrage épais et son parement: Application aux enceintes de confinement.* PhD thesis, Université Grenoble Alpes (ComUE), 2016.

[Bou18] David E-M Bouhjiti. *Analyse probabiliste de la fissuration et du confinement des grands ouvrages en béton armé et précontraint.* PhD thesis, Université Grenoble Alpes, 2018.

[BTT24] Nicolas Barral, Tommaso Taddei, and Ishak Tifouti. Registration-based model reduction of parameterized pdes with spatio-parameter adaptivity. *Journal of Computational Physics*, 499:112727, 2024.

[Bye99] Gerry C Bye. *Portland cement: composition, production and properties.* Thomas Telford, 1999.

[CA19] Fabien Casenave and Nissrine Akkari. An error indicator-based adaptive reduced order model for nonlinear structural mechanics—application to high-pressure turbine blades. *Mathematical and computational applications*, 24(2):41, 2019.

[CAB+20] Fabien Casenave, Nissrine Akkari, Felipe Bordeu, Christian Rey, and David Ryckelynck. A nonintrusive distributed reduced-order modeling framework for nonlinear structural mechanics—Application to elastoviscoplastic computations. *International journal for numerical methods in engineering*, 121(1):32–53, 2020.

[CACF17] Todd Chapman, Philip Avery, Pat Collins, and Charbel Farhat. Accelerated mesh sampling for the hyper reduction of nonlinear computational models. *International Journal for Numerical Methods in Engineering*, 109(12):1623–1654, 2017.

[CD15] Albert Cohen and Ronald DeVore. Approximation of high-dimensional parametric PDEs. *Acta Numerica*, 24:1–159, 2015.

[CD16] Albert Cohen and Ronald DeVore. Kolmogorov widths under holomorphic mappings. *IMA Journal of Numerical Analysis*, 36(1):1–12, 2016.

[CFCA13] Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amsallem. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623–647, 2013.

[CGT00] Andrew Conn, Nick Gould, and Philippe Toint. Trust regions methods. Society for Industrial and Applied Mathematics. 2000.

[Che11] Wei Chen. *Etude expérimentale de la perméabilité du béton sous conditions thermiques et hydriques variables*. PhD thesis, Ecole centrale de Lille, 2011.

[DCAR20] Thomas Daniel, Fabien Casenave, Nissrine Akkari, and David Ryckelynck. Model order reduction assisted by deep neural networks (ROM-net). *Advanced Modeling and Simulation in Engineering Sciences*, 7:1–27, 2020.

[DY22] Eugene Du and Masayuki Yano. Efficient hyperreduction of high-order discontinuous Galerkin methods: element-wise and point-wise reduced quadrature formulations. *Journal of Computational Physics*, 466:111399, 2022.

[EDF24] Electricité de France EDF. Finite element code_aster, Analysis of Structures and Thermomechanics for Studies and Research. *Open source on www.code-aster.org*, 1989-2024.

[ES95] Richard Everson and Lawrence Sirovich. Karhunen–Loeve procedure for gappy data. *JOSA A*, 12(8):1657–1664, 1995.

[FACC14] Charbel Farhat, Philip Avery, Todd Chapman, and Julien Cortial. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering*, 98(9):625–662, 2014.

[FCA15] Charbel Farhat, Todd Chapman, and Philip Avery. Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *International journal for numerical methods in engineering*, 102(5):1077–1110, 2015.

[FDM21] Stefania Fresca, Luca Dede', and Andrea Manzoni. A comprehensive deep learning-based approach to reduced order modeling of nonlinear time-dependent parametrized PDEs. *Journal of Scientific Computing*, 87:1–36, 2021.

[FK09] Alexander IJ Forrester and Andy J Keane. Recent advances in surrogate-based optimization. *Progress in aerospace sciences*, 45(1-3):50–79, 2009.

[Fle00] Roger Fletcher. *Practical methods of optimization.* John Wiley & Sons, 2000.

[FM22] Stefania Fresca and Andrea Manzoni. POD-DL-ROM: Enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 388:114181, 2022.

[FMPG12] A Foucault, S Michel-Ponnelle, and E Galenne. A new creep model for NPP containment behaviour prediction. In *International conference on Numerical modeling Strategies for sustainable concrete structures*, 2012.

[Fou88] Jean Baptiste Joseph Fourier. *Théorie analytique de la chaleur.* Gauthier-Villars et fils, 1888.

[FS03] Marco Fahl and Ekkehard W Sachs. Reduced order modelling approaches to PDE-constrained optimization based on proper orthogonal decomposition. In *Large-scale PDE-constrained optimization*, pages 268–280. Springer, 2003.

[GFTBM21] Sebastian Grimberg, Charbel Farhat, Radek Tezaur, and Charbel Bou-Mosleh. Mesh sampling and weighting for the hyperreduction of nonlinear Petrov–Galerkin reduced-order models with local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 122(7):1846–1874, 2021.

[GMW19] Philip E Gill, Walter Murray, and Margaret H Wright. *Practical optimization.* SIAM, 2019.

[Gra95] Laurent Granger. *Comportement différé du béton dans les enceintes de centrales nucléaires: analyse et modélisation.* PhD thesis, Ecole Nationale des ponts et Chaussées, 1995.

[Gun02] Max D Gunzburger. *Perspectives in flow control and optimization.* SIAM, 2002.

[Haa13] Bernard Haasdonk. Convergence rates of the pod–greedy method. *ESAIM: Mathematical modelling and numerical Analysis*, 47(3):859–873, 2013.

[Haa17] Bernard Haasdonk. Reduced basis methods for parametrized PDEs–a tutorial introduction for stationary and instationary problems. *Model reduction and approximation: theory and algorithms*, 15:65, 2017.

[Had02] Jacques Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University bulletin*, pages 49–52, 1902.

[HCF17] Joaquin Alberto Hernandez, Manuel Alejandro Caicedo, and Alex Ferrer. Dimensional hyper-reduction of nonlinear finite element models via empirical cubature. *Computer methods in applied mechanics and engineering*, 313:687–722, 2017.

[HLR18] Christian Himpe, Tobias Leibner, and Stephan Rave. Hierarchical approximate proper orthogonal decomposition. *SIAM Journal on Scientific Computing*, 40(5):A3267–A3292, 2018.

[HO08] Bernard Haasdonk and Mario Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.

[HRS$^+$16] Jan S Hesthaven, Gianluigi Rozza, Benjamin Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.

[HS87] Bernard Halphen and Jean Salençon. *Elasto-plasticité - Cours de calcul des structures anélastiques.* 1987.

[IST22] Angelo Iollo, Giulia Sambataro, and Tommaso Taddei. An adaptive projection-based model reduction method for nonlinear mechanics with internal variables: Application to thermo-hydro-mechanical systems. *International Journal for Numerical Methods in Engineering*, 123(12):2894–2918, 2022.

[Jam88] Antony Jameson. Aerodynamic design via control theory. *Journal of scientific computing*, 3:233–260, 1988.

[Joh12] Steven G Johnson. Notes on adjoint methods for 18.335. *Introduction to Numerical Methods*, 2012.

[JPCGH07] Ludovic Jason, Gilles Pijaudier-Cabot, Shahrokh Ghavamian, and Antonio Huerta. Hydraulic behaviour of a representative structural volume for containment buildings. *Nuclear engineering and design*, 237(12-13):1259–1274, 2007.

[Kal60] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[KO24] Tim Keil and Mario Ohlberger. A relaxed localized trust-region reduced basis approach for optimization of multiscale problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 58(1):79–105, 2024.

[Koa23] Herman Koala. *Loi de comportement du béton VeRCoRs en conditions accidentelles.* PhD thesis, Université Paul Sabatier-Toulouse III, 2023.

[Kol] Andreĭ Kolmogoroff. Uber die beste Annaherung von Funktionen einer gegebenen Funktionenklasse.

[LC04] Henri Le Chatelier. *Recherches expérimentales sur la constitution des mortiers hydrauliques.* Dunod, 1904.

[LDT86] François-Xavier Le Dimet and Olivier Talagrand. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A: Dynamic Meteorology and Oceanography*, 38(2):97–110, 1986.

[LH95] Charles L Lawson and Richard J Hanson. *Solving least squares problems.* SIAM, 1995.

[LL17] Hans Petter Langtangen and Anders Logg. *Solving PDEs in python: the FEniCS tutorial I.* Springer Nature, 2017.

[LMQR13] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Generalized reduced basis methods and n-width estimates for the approximation of the solution manifold of parametric PDEs. *Analysis and numerics of partial differential equations*, pages 307–329, 2013.

[LMRC21] Kjetil O Lye, Siddhartha Mishra, Deep Ray, and Praveen Chandrashekar. Iterative surrogate model optimization (ISMO): An active learning algorithm for PDE constrained optimization with deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 374:113575, 2021.

[Lor86] Andrew C Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194, 1986.

[LT06] Patrick Le Tallec. Mécanique des milieux continus. *Editions de l'École polytechnique*, 2006.

[MAA88] R Mensi, P Acker, and A Attolou. Séchage du béton: analyse et modélisation. *Materials and structures*, 21:3–12, 1988.

[MBC00] Michael D McKay, Richard J Beckman, and William J Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61, 2000.

[MM13] Yvon Maday and Olga Mula. A generalized empirical interpolation method: application of reduced basis techniques to data assimilation. In *Analysis and numerics of partial differential equations*, pages 221–235. Springer, 2013.

[Moo81] Bruce Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE transactions on automatic control*, 26(1):17–32, 1981.

[MPT02] Yvon Maday, Anthony T Patera, and Gabriel Turinici. A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. *Journal of Scientific Computing*, 17:437–446, 2002.

[MR22] Johannes Maierhofer and Daniel J Rixen. Model order reduction using hyperreduction methods (DEIM, ECSW) for magnetodynamic FEM problems. *Finite Elements in Analysis and Design*, 209:103793, 2022.

[MZ93] Stéphane G Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on signal processing*, 41(12):3397–3415, 1993.

[N$^+$18] Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

[NM65] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[PÁKRR21] Pablo Pereira Álvarez, Pierre Kerfriden, David Ryckelynck, and Vincent Robin. Real-time data assimilation in welding operations using thermal imaging and accelerated high-fidelity digital twinning. *Mathematics*, 9(18):2263, 2021.

[Pow64] Michael JD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal*, 7(2):155–162, 1964.

[QGVW17] Elizabeth Qian, Martin Grepl, Karen Veroy, and Karen Willcox. A certified trust region reduced basis approach to PDE-constrained optimization. *SIAM Journal on Scientific Computing*, 39(5):S434–S460, 2017.

[QMN15] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.

[Qua17] Alfio Quarteroni. *Numerical models for differential problems.* Springer, 2017.

[Règ92] BPEL Règles. 91 règles techniques de conception et de calcul des ouvrages et constructions en béton précontraint suivant la méthode des états limites. *Edition Eyrolles*, 1992.

[RHP08] Gianluigi Rozza, Dinh Bao Phuong Huynh, and Anthony T Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

[Ros22] Donatien Rossat. *Quantification d'incertitudes inverse par inférence Bayésienne pour des modèles multi-physiques de grands ouvrages de confinement.* PhD thesis, Université Grenoble Alpes, 2022.

[Ryc05] David Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of computational physics*, 202(1):346–366, 2005.

[Sal05] Jean Salençon. *Mécanique des milieux continus: Concepts généraux*, volume 1. Editions Ecole Polytechnique, 2005.

[Sch10] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.

[Sir87] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.

[T$^+$97] Harry FW Taylor et al. *Cement chemistry*, volume 2. Thomas Telford London, 1997.

[Tal97] Olivier Talagrand. Assimilation of observations, an introduction (gtspecial issuelt-data assimilation in meteology and oceanography: Theory and practice). *Journal of the Meteorological Society of Japan. Ser. II*, 75(1B):191–209, 1997.

[TFPV92] Saul A Teukolsky, Brian P Flannery, WH Press, and WT Vetterling. Numerical recipes in C. *SMR*, 693(1):59–70, 1992.

[Tik] Andreĭ Tikhonov. Solutions of ill-posed problems.

[Tro04] Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information theory*, 50(10):2231–2242, 2004.

[VG80] M Th Van Genuchten. A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil science society of America journal*, 44(5):892–898, 1980.

[VGO$^+$20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[Vol11] Stefan Volkwein. Model reduction using proper orthogonal decomposition. *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http://www. uni-graz. at/imawww/volkwein/POD. pdf*, 1025, 2011.

[VPRP03] Karen Veroy, Christophe Prud'Homme, Dimitrios Rovas, and Anthony Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference*, page 3847, 2003.

[W$^+$63] Mark L Wilkins et al. *Calculation of elastic-plastic flow.* University of California, Ernest L. Lawrence Radiation Laboratory; Reproduced by National Technical Information Service, Spirngfield Va., 1963.

[WAZF12] Kyle Washabaugh, David Amsallem, Matthew Zahr, and Charbel Farhat. Nonlinear model reduction for CFD problems using local reduced-order bases. In *42nd AIAA Fluid Dynamics Conference and Exhibit*, page 2686, 2012.

[WNLDZ92] Zhi Wang, I Michael Navon, François-Xavier Le Dimet, and X Zou. The second order adjoint analysis: theory and applications. *Meteorology and atmospheric physics*, 50:3–20, 1992.

[Wri06] Stephen J Wright. *Numerical optimization.* 2006.

[WZ23] Tianshu Wen and Matthew J Zahr. A globally convergent method to accelerate large-scale optimization using on-the-fly model hyperreduction: application to shape optimization. *Journal of Computational Physics*, 484:112082, 2023.

[XBMJ94] Yunping Xi, Zdeněk P Bažant, Larissa Molina, and Hamlin M Jennings. Moisture diffusion in cementitious materials moisture capacity and diffusivity. *Advanced Cement Based Materials*, 1(6):258–266, 1994.

[YHZ21] Masayuki Yano, Tianci Huang, and Matthew J Zahr. A globally convergent method to accelerate topology optimization using on-the-fly model reduction. *Computer Methods in Applied Mechanics and Engineering*, 375:113635, 2021.

[YM13] Yao Yue and Karl Meerbergen. Accelerating optimization of parametric linear systems by model order reduction. *SIAM Journal on Optimization*, 23(2):1344–1370, 2013.

[YP19] Masayuki Yano and Anthony T Patera. An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs. *Computer Methods in Applied Mechanics and Engineering*, 344:1104–1123, 2019.

[YWD15] Mehrdad Yaghoobi, Di Wu, and Mike E Davies. Fast non-negative orthogonal matching pursuit. *IEEE Signal Processing Letters*, 22(9):1229–1233, 2015.

[Zah16] Matthew Joseph Zahr. *Adaptive model reduction to accelerate optimization problems governed by partial differential equations.* PhD thesis, Stanford University, 2016.

[ZBLN97] Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on mathematical software (TOMS)*, 23(4):550–560, 1997.

[ZCK19] Matthew J Zahr, Kevin T Carlberg, and Drew P Kouri. An efficient, globally convergent method for optimization under uncertainty using adaptive model reduction and sparse grids. *SIAM/ASA Journal on Uncertainty Quantification*, 7(3):877–912, 2019.

[ZF15] Matthew J Zahr and Charbel Farhat. Progressive construction of a parametric reduced-order model for PDE-constrained optimization. *International Journal for Numerical Methods in Engineering*, 102(5):1111–1135, 2015.