



HAL
open science

Développement de méthodes d'apprentissage profond pour la reconstruction de données spatio-temporelles hautement comprimées : application aux signaux des réseaux de capteurs sans fil et aux simulations céramiques de structures colloïdales

Sébastien Rousseau

► To cite this version:

Sébastien Rousseau. Développement de méthodes d'apprentissage profond pour la reconstruction de données spatio-temporelles hautement comprimées : application aux signaux des réseaux de capteurs sans fil et aux simulations céramiques de structures colloïdales. Réseau de neurones [cs.NE]. Université de Limoges, 2024. Français. NNT : 2024LIMO0021 . tel-04794391

HAL Id: tel-04794391

<https://theses.hal.science/tel-04794391v1>

Submitted on 20 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université de Limoges

ED 653 – Sciences et Ingénierie (SI)

Institut de Recherche XLIM & Institut de Recherche sur les Céramiques IRCER

Thèse pour obtenir le grade de

Docteur de l'Université de Limoges

Sciences et Ingénierie pour l'information mention image, signal et automatique

Présentée et soutenue par

Sébastien ROUSSEAU

Le 2 Mai 2024

DÉVELOPPEMENT DE MÉTHODES D'APPRENTISSAGE PROFOND POUR LA RECONSTRUCTION DE DONNÉES SPATIO-TEMPORELLES HAUTEMENT COMPRIMÉES – APPLICATION AUX SIGNAUX DES RÉSEAUX DE CAPTEURS SANS FIL ET AUX SIMULATIONS CÉRAMIQUES DE STRUCTURES COLLOÏDALES

Thèse dirigée par Raymond QUÉRÉ, Romain NÉGRIER et Fabien COURRÈGES

JURY :

Président du jury

M. Guillaume ANDRIEUX, Professeur des Universités – IETR – Université de Nantes

Rapporteurs

M. Philippe OWEZARSKI, Directeur de Recherche CNRS – LAAS – Université P.Sabatier Toulouse

M. Abbas BRADAI, Professeur des Universités – LEAT – Université de côte d'Azur

Examineurs

M. Arnaud VIDECOQ, Professeur des Universités – IRCER – Université de Limoges

M. Raymond QUÉRÉ, Professeur des Universités Émérite – XLIM – Université de Limoges

M. Romain NÉGRIER, Maître de Conférence – XLIM – Université de Limoges

Invités

M. Fabien COURRÈGES, Maître de Conférence – XLIM – Université de Limoges

À ma famille. À Kanya.

Un intellectuel est quelqu'un qui se mêle de ce qui ne le regarde pas.

Jean-Paul Sartre

Remerciements

Mes premiers remerciements vont à mes encadrants Fabien Courrèges, Romain Négrier et Raymond Quéré qui sont toujours restés disponibles, à l'écoute, de bons conseils, solidaires. Je les remercie pour le cadre de travail agréable qu'ils m'ont donné et pour tous nos échanges. Je leur dois un parcours de thèse riche et stimulant. Je les remercie également d'avoir imaginé et porté ce projet de thèse. Ce fut un plaisir et un honneur de les avoir eu comme directeurs de thèse.

Pour le sérieux et la bonne ambiance dans les locaux d'XLIM Brive-la-Gaillarde, je remercie aussi mes collègues Florent, Marie, Mohamed, Anass, Raphaël, Fabien, Jean-Christophe, Michel, Adèle, Bernadette, Anne-Marie, Abdel, Paul, Souhail, Jose Anderson, Ndoua Luc Arnaud, Sylvie, Alexandre, Julien et toute l'équipe d'enseignants et d'enseignant-chercheurs de l'IUT GEII de Brive-la-Gaillarde. J'ai un remerciement tout particulier pour Olivier Halluin et Michèle Lalande sans qui mon recrutement pour cette thèse n'aurait pas eu lieu.

Je remercie Arnaud Videcoq de l'IRCER pour son expertise, sa sympathie et son implication dans les travaux de thèse relatifs au chapitre III.

Je remercie les responsables du LabEx Σ -LIM pour avoir co-financé mon contrat doctoral et soutenu ce projet.

Je remercie Frank Itoua-Engoti pour m'avoir partagé son travail de thèse, me permettant ainsi d'avancer rapidement dans la compréhension de mon sujet et dans la préparation des résultats comparatifs.

Enfin, je remercie ma famille et mes amis pour avoir suivi ma thèse avec intérêt et m'avoir soutenu. J'ai pu compter sur chacun d'eux pendant les moments clefs de mon travail qui nécessitait aussi de leur accorder moins de mon temps. Je remercie ma compagne, Adèle, pour sa patience et son soutien au quotidien. Je remercie ma mère, Valérie, qui m'a suivi et appuyé dans mes choix.

Droits d'auteurs

Cette création est mise à disposition selon le Contrat :

« Attribution-Pas d'Utilisation Commerciale-Pas de modification 3.0 France »

disponible en ligne : <http://creativecommons.org/licenses/by-nc-nd/3.0/fr/>



Table des matières

Table des figures	9
Table des tableaux	12
Table des algorithmes	13
Liste des acronymes	14
Introduction générale	17
Contexte et motivations de la thèse	18
Problématiques et objectifs	20
Plan	20
I Etat de l'art	22
I.1 Connaissances fondamentales	24
I.1.1 Les réseaux de capteurs sans fil	24
I.1.2 L'acquisition comprimée	27
I.1.3 L'apprentissage profond	34
I.1.4 Les classes de réseaux de neurones	39
I.2 Apprentissage profond en lien avec les objectifs de la thèse	45
I.2.1 Des modèles de réseaux de neurones	45
I.2.2 Préviation des données	50
I.2.3 Imputation de données manquantes	53
II Développement de méthodes de reconstruction des signaux comprimés par apprentissage profond – application au réseau de capteurs	58
II.1 Méthodologie	60
II.1.1 Méthodes de reconstruction	60
II.1.2 Les algorithmes de traitement du signal	73
II.1.3 Création d'une API dédiée	77

II.2 Applications	81
II.2.1 Les ressources d'expérimentation	81
II.2.2 Application de la méthode temporelle	87
II.2.3 Application de la méthode tempo-spatiale	94
II.3 Bilan	110
III Contribution à l'identification de modèles de simulations de structures colloïdales	113
III.1 Les simulations de structures colloïdales	115
III.1.1 Les activités de l'IRCER	115
III.1.2 Le potentiel d'interaction de particules Janus en suspension . . .	121
III.1.3 L'identification paramétrique	126
III.2 Application	129
III.2.1 Développement des modèles	129
III.2.2 Tests	134
III.3 Bilan	144
Conclusion générale	145
Bilan général	146
Contributions	147
Perspectives	148
A Annexes	150
A.1 API deepsignals	151
A.1.1 Diagramme de paquetages	151
A.1.2 Diagramme de classes	152
A.1.3 Diagramme de cas d'utilisation	154
A.2 Jeu de données	156
A.2.1 Signaux d'un réseau de capteur sans fil	156
B Bibliographie	161
Références	162
Liste des travaux	173

Table des figures

I.1	Topologies des réseaux de capteurs sans fil	25
I.2	Ensemble des mesures des capteurs placés au 3ème étage de l'IUT GEII de Brive-la-Gaillarde entre le 20/09/2016 et le 21/09/2016 à 23h59 (figure issue de [9])	31
I.3	Spectre du vecteur champ spatio-temporel obtenu par la transformation \mathcal{W} (figure issue de [9])	32
I.4	Neurone artificiel - Perceptron	35
I.5	Ajustement des poids neuronaux par rétropropagation	36
I.6	Phénomènes d'ajustements au jeu de données d'entraînement	36
I.7	Représentation d'un gradient sur une courbe d'erreur pour un échantillon de donnée	38
I.8	Réseau de neurones à propagation avant	39
I.9	Neurone artificiel récurrent	39
I.10	Cellule récurrente Long Short-Term Memory	40
I.11	Couche de convolution	42
I.12	Les fonctions d'activations les plus communes	43
I.13	Modèle simplifié d'auto-encodeur	46
I.14	Modèle simplifié de réseau antagoniste génératif	46
I.15	Modélisation simplifiée du Transformeur	48
II.1	Théorie de l'optimisation d'une matrice de détection aléatoire d'Acquisition Comprimée (AC) à l'aide d'Apprentissage Profond (AP)	61
II.2	Illustration d'un tenseur 3D pour les signaux d'un RCSF	63
II.3	Illustration d'un tenseur 5D pour les signaux d'un RCSF	63
II.4	Prévision d'un signal: marge d'erreur croissante dans le temps	66
II.5	Reconstruction temporelle collaborative causale et anti-causale	67
II.6	Reconstruction spatiale réalisée par auto-encodage	68
II.7	Exemples de mise en difficulté de chacun des Auto-Encodeurs (AEs)	70

II.8 Schéma d'une itération de reconstruction tempo-spatiale des signaux compressés d'un Réseau de Capteurs Sans Fil (RCSF)	71
II.9 Illustration du post-traitement d'un signal prédit dans l'intervalle de deux valeurs réelles	75
II.10 Station de base Meshlium	82
II.11 Noeud de capteurs Waspnote	82
II.12 Capteur de température et d'humidité relative DHT22	83
II.13 Exemple de signaux de températures des noeuds de capteurs 002 et 007 du rez-de-chaussée du bâtiment GEII de Brive-la-Gaillarde	84
II.14 Synchronisation des horodatages des signaux du RCSF	84
II.15 Erreur moyenne de reconstruction des signaux de températures par interpolation spline en fonction du taux de compression	85
II.16 Interpolation des valeurs des signaux du RCSF	86
II.17 Modèle de prédicteur utilisé pour l'application de la méthode temporelle	87
II.18 Erreur moyenne de reconstruction de signaux de températures par taux de compression	89
II.19 Exemple de reconstruction d'un signal test compressé à 50%	90
II.20 Exemple de reconstruction d'un signal test compressé à 80%	91
II.21 Exemple de reconstruction d'un signal test compressé à 90%	92
II.22 Modèle de prédicteur utilisé pour l'application de la méthode tempo-spatiale	94
II.23 Modèle d'AE à réduction de dimensionalité pour cette application (AEDR)	95
II.24 Modèle d'AE à large spectre pour cette application (AEBS)	96
II.25 Modèle d'AE variationnel pour cette application (VAE)	96
II.26 Modèle de discriminateur pour cette application	97
II.27 Exemple de reconstructions spatiales à un instant t par: autoencodeur à réduction de dimensionalité (AEDR), autoencodeur à large spectre (AEBS), autoencodeur variationnel (VAE)	100
II.28 Erreur moyenne de reconstructions spatiales de signaux de températures par taux de compression	101
II.29 Erreur moyenne de reconstruction de signaux de températures par vraisemblance	102
II.30 Comparaison des erreurs moyenne de reconstructions spatiale et temporelle de signaux de températures par taux de compression	102
II.31 Erreur moyenne de reconstruction de signaux de températures par fiabilité statistique de prédiction	103

II.32	Erreur moyenne de reconstructions de signaux de températures par taux de compression	104
II.33	Exemple de reconstruction d'un signal test compressé à 97,2%	104
II.34	Exemple de reconstructions de signaux tests compressés à 99% et 99,1%	105
II.35	Exemple de reconstructions de signaux tests compressés à 99,3% et 99,4%	106
II.36	Exemple de reconstructions de signaux tests compressés à 99,6% et 99,7%	107
II.37	Exemple de reconstructions de signaux tests compressés à 99,9%	108
II.38	Exemple de reconstruction d'un signal test compressé à 100%	109
III.1	Modélisation de nanoparticules Janus (particules à deux faces) en suspension dans un milieu de dispersion (figure issue de [101])	117
III.2	Modélisation d'agrégats de particules dans une suspension colloïdale (figure issue de [101])	119
III.3	Modélisation d'une particule Janus	121
III.4	Modélisation de deux particules Janus en interaction	122
III.5	Modélisation d'une particule framboise (figure issue de [101])	125
III.6	Similitudes entre identification et apprentissage	126
III.7	Modèle simulateur	130
III.8	Modèle en boîte grise pour l'IRCER	131
III.9	Modèle de générateur de paramètres pour les formules de l'IRCER	132
III.10	Modèle final de générateur de paramètres pour les formules de l'IRCER	133
III.11	Modèle pour la preuve de concept	135
III.12	Modèle de générateur de paramètre avec entraînement local	137
A.1	Diagramme UML de paquetages de l'IPA deepsignals	151
A.2	Diagramme UML de classes de l'IPA deepsignals	153
A.3	Diagramme UML de cas d'utilisation de l'IPA deepsignals	155
A.4	Signaux de températures des noeuds de capteur du rez-de-chaussée du bâtiment GEII de Brive-la-Gaillarde	156
A.5	Signaux de températures des noeuds de capteur du 1er étage du bâtiment GEII de Brive-la-Gaillarde	157
A.6	Signaux de températures des noeuds de capteur du 2ème étage du bâtiment GEII de Brive-la-Gaillarde	159
A.7	Signaux de températures des noeuds de capteur du 3ème étage du bâtiment GEII de Brive-la-Gaillarde	160

Table des tableaux

I.1	Comparaison de performances pour la régression sur le jeu de données Physionet en fonction du taux de données manquantes (en racine d'erreur quadratique moyenne)	55
I.2	Comparaison de performances pour la régression sur le jeu de données Physionet en fonction du taux de données manquantes (en erreur moyenne absolue)	55
I.3	Comparaison de performances de prédiction de mortalité sur le jeu de donnée Physionet (en score AUC)	55
II.1	Caractéristiques du capteur DHT22	83
II.2	Fiabilité statistique pour différents horizons de prédictions de l'application (exprimés en nombre de pas de temps)	88
II.3	Temps de calculs de reconstruction du jeu de données entier d'un période de 9 mois (33 signaux de longueur 4889)	92
II.4	Consommation énergétique du matériel de cette application	93
II.5	Fiabilité statistique pour différents horizons de prédictions de l'application (exprimés en nombre de pas de temps)	99
III.1	Comparatif des caractéristiques d'IP et d'AP	127
III.2	Échantillon de données pour une particule	129
III.3	Résultats d'entraînement par nombre de cellules pour le modèle IRCER	138
III.4	Résultats d'entraînement par bruit de l'Evaluator pour le modèle IRCER	139
III.5	Résultats d'apprentissage par configuration d'entraînement pour le modèle IRCER	139
III.6	Résultats d'apprentissage par temps de simulation	141
III.7	Résultats d'apprentissage pour différents volumes de neurones à 0.4 secondes de temps de simulation	142

Table des algorithmes

II.1	Calcul des fiabilités statistiques des prédictions	73
II.2	Post-traitement d'un signal prédit	74
II.3	Ajustement d'un signal à un signal discriminant	76

Liste des acronymes

- AA** – Apprentissage Automatisé.
- AC** – Acquisition Comprimée.
- ACD** – Acquisition Comprimée Distribuée.
- ACK** – Acquisition Comprimée de Kronecker.
- AE** – Auto-Encodeur.
- AP** – Apprentissage Profond.
- API** – Application Programming Interface.
- ARIMA** – AutoRegressive Integrated Moving Average.
- ARX** – Auto Regressive model with eXternal inputs.
- CNN** – Convolutional Neural Network.
- CS** – Compressed Sensing.
- DCS** – Distributed Compressive Sensing.
- DL** – Deep Learning.
- DVS** – Décomposition en Valeurs Singulières.
- FARIMA** – Fractionnaly AutoRegressive Integrated Moving Average.

- FNN** – Feed-forward Neural Network.
- GAN** – Generative Adversarial Network.
- GNN** – Graph Neural Network.
- GPU** – Graphics Processing Unit.
- GRU** – Gated Recurrent Unit.

- HR** – Humidité Relative.

- IA** – Intelligence Artificielle.
- IdO** – Internet des Objets.
- IEEE** – Institute of Electrical and Electronics Engineers.
- IoT** – Internet of Things.
- IP** – Identification Paramétrique.
- IPA** – Interface de Programmation d'Application.
- IRCER** – Institut de Recherche sur les Céramiques IRCER.

- JSM** – Joint Sparsity Model.

- KCS** – Kronecker Compressive Sensing.

- LSTM** – Long Short-Term Memory.

- MAE** – Mean Absolute Error.
- ML** – Machine Learning.
- MP** – Matching Pursuit.
- MSE** – Mean Squared Error.

- NN** – Neural Network.

- PReLU** – Parametric Rectified Linear Unit.

- RAG** – Réseau Antagoniste Génératif.
- RCSF** – Réseau de Capteurs Sans Fil.
- RN** – Réseau de Neurones.
- RNC** – Réseau de Neurones Convolutifs.
- RNG** – Réseau de Neurones Graphique.
- RNN** – Recurrent Neural Network.
- RNPA** – Réseau de Neurones à Propagation Avant.
- RNR** – Réseau de Neurones Récurents.

- Seq2seq** – Sequence-to-sequence.
- SRF** – Systèmes Radio-Fréquences.
- SRI** – Systèmes et Réseaux Intelligents.
- ST** – Série Temporelle.
- ST-LSTM** – Spatio-Temporal Long Short-Term Memory.

- TCD** – Transformée en Cosinus Discrète.

- UML** – Unified Modeling Language.
- UTG** – Unité de Traitement Graphique.

- WSN** – Wireless Sensor Network.

- XLIM** – Institut de Recherche pluridisciplinaire XLIM.

Introduction générale

Sommaire

Contexte et motivations de la thèse	18
Problématiques et objectifs	20
Plan	20

Le verbe *comprimer* est issu du verbe latin *premere* dont la forme nominale est *pressum* d'où provient le mot *compression*. Le verbe *compresser* naît d'un anglicisme du verbe *compress* qui se traduit par *comprimer*. Son usage est principalement propre au domaine de l'informatique (exemple: "Compresser un dossier"). Dans le contexte de la thèse, les deux termes sont utilisés et ont le sens de: réduire le volume.

Contexte et motivations de la thèse

Cette thèse s'inscrit dans un contexte technologique récent et un projet de continuité de recherches au sein de l'Institut de Recherche pluridisciplinaire XLIM (XLIM) sur le site de Brive-la-Gaillarde. Ses enjeux sont économiques et écologiques. Elle est portée par le LabEx Σ -LIM et la collaboration entre l'Institut de Recherche sur les Céramiques IRCER (IRCER), l'axe Systèmes Radio-Fréquences (SRF) et l'axe Systèmes et Réseaux Intelligents (SRI) d'XLIM. Il s'agit au départ de développer et d'employer des solutions d'apprentissage profond dans une nouvelle approche de type co-conception d'un couple d'entités, l'une étant un générateur de données, l'autre étant un système dynamique alimenté et contraint par les données et dont la structure interne doit être énergétiquement optimisée. Afin d'évaluer et de démontrer la puissance et la versatilité de ce cadre théorique général, deux contextes d'application permettent de développer des formes plus spécifiques à cette théorie.

Le premier contexte d'application est celui des objets connectés. Des domaines industriels aux fournisseurs d'énergies, à la santé, aux bâtiments intelligents, jusqu'aux particuliers, les objets connectés se sont immiscés partout dans notre environnement en amenant avec eux leurs problématiques de consommation d'énergie, de maintenance, de saturation des bandes radio et de stockage massif de données. Cette application poursuit des travaux afférents à l'AC, ou Compressed Sensing (CS), menés par Frank Itoua-Engoti durant sa thèse intitulée « Réalisation d'une plate-forme pour l'optimisation de réseaux de capteurs sans fil appliqués au bâtiment intelligent » et dirigée par Raymond Quéré, Michèle Lalande et Fabien Courrèges. Les avantages de l'AC sont multiples, elle peut réduire l'acquisition des données, diminuer les coûts de communication, prévenir l'agrégation de données sur les bandes radio et minimiser les volumes de stockage des données dans le cloud. Les travaux de recherches présentés dans ce manuscrit exploitent un jeu de données de signaux de RCSF acquis sur le campus de Brive-la-Gaillarde. Le RCSF avait été installé par M. Itoua-Engoti en 2018.

Le second contexte d'application est propre aux simulations de structures colloïdales des chercheurs de l'IRCER qui recherchent de nouveaux modèles de calculs pour remplacer ou assister les leurs qui sont extrêmement coûteux en temps et en mémoire bien qu'exploitant la puissance de calcul des cartes graphiques. Ce genre de simulation soumis au mouvement brownien est très complexe à calibrer car beaucoup de paramètres varient selon le nombre d'interactions quantiques en jeu. La difficulté majeure est de trouver les modèles mathématiques qui reproduisent au mieux ces dynamiques de particules dans une littérature plurivalente.

Le monde de la recherche est profondément modifié par l'émergence d'une nouvelle technologie depuis le début des années 2000 : l'AP, ou Deep Learning (DL). L'AP est une branche de l'Apprentissage Automatisé (AA), ou Machine Learning (ML), qui est elle-même une branche de l'Intelligence Artificielle (IA). Pour en faire une brève définition : une IA désigne tout processus d'automatisation de tâches intellectuelles normalement effectuées par des humains. Ce qui distingue l'AA de l'IA est la notion d'apprentissage par laquelle une structure statistique s'entraîne au travers de données à automatiser une tâche. Enfin, l'AP concerne uniquement les modèles de couches hiérarchiques que l'on appelle les Réseaux de Neurones (RNs), ou Neural Networks (NNs). L'AP trouve des applications dans tous les domaines qui impliquent l'étude de données : la santé, l'industrie, les réseaux sociaux, la finance, le commerce, l'écologie, la physique, l'audiovisuel, la linguistique, l'archéologie, l'immobilier; cette liste n'est pas exhaustive.

Le projet de thèse s'articule autour de l'idée que l'AA puisse, à travers l'AP, optimiser, améliorer ou supplanter l'AC dans le cadre des signaux de RCSF. La nouvelle méthodologie proposée permet de repousser les performances d'économie et d'autonomie énergétique des nœuds d'un RCSF et ouvre la voie à de nouvelles applications dans l'Internet des Objets (IdO), ou Internet of Things (IoT). Toujours motivé par des objectifs visant à réduire des coûts, l'approche utilisée pour les RCSFs est étudiée pour la simulation de structures colloïdales. L'AP peut apporter des solutions aux problématiques des objets connectés et des simulations d'interactions de particules en assistant et paramétrant des modèles de calculs tel que l'AC ou la dynamique de Langevin, cette thèse vise à explorer ses possibilités.

Problématiques et objectifs

Les problématiques de la thèse découlent du projet, du contexte, des outils technologiques récents, des besoins identifiés par le laboratoire XLIM et l'IRCER :

- L'acquisition temps réel de signaux non comprimés consomment de l'énergie.
- La maintenance d'un système d'objets connectés est coûteuse et complexe à mesure que le nombre d'objets augmente.
- Les consommations de temps de calculs et de mémoire des algorithmes de reconstruction de l'AC et des simulations de structures colloïdales sont excessives et compliquent leur utilisation pour des applications pratiques.
- Les méthodes d'AC présentent des erreurs de reconstruction élevées à un fort taux de compression des signaux.
- Les modèles de calculs de mouvements de particules en interaction et soumises au mouvement brownien sont nombreux et approximatifs.

Les objectifs répondent aux problématiques et présentent aussi les ambitions du travail de recherche :

- Concevoir une méthodologie à l'état de l'art de compression et de reconstruction de signaux basée sur l'AP, qu'elle soit implémentable facilement et moins gourmande en mémoire et en temps de calculs que les algorithmes de reconstruction d'AC.
- Obtenir des reconstructions de signaux hautement comprimés plus proche de la réalité qu'avec les méthodes d'AC.
- Employer des techniques d'AP pour identifier une alternative relativement plus simple aux modèles actuels de simulations de structures colloïdales de l'IRCER qui décrit fidèlement les propriétés des matériaux céramiques en production.

Plan

Le chapitre I constitue l'état de l'art. Il rappelle les connaissances fondamentales nécessaires à la compréhension de la thèse et de son contexte puis il présente les travaux récents menés dans le domaine de l'AP en lien avec les objectifs de la thèse. Les

trois grands thèmes sont: les réseaux de capteurs sans fil et leurs signaux; l'acquisition comprimée, ses théories et ses applications; l'apprentissage profond, des bases aux dernières innovations en matière de traitement des données temporelles et spatio-temporelles. La littérature relative au traitement des données temporelles et spatio-temporelles par AP est vaste, les applications et les approches différentes sont très nombreuses et sans cesse renouvelées. Il n'existe pas de méthode fédératrice pour réaliser les objectifs du projet de thèse, c'est pourquoi ce chapitre est aussi une étude de pertinence au regard des objectifs de la thèse afin de placer ses travaux au niveau de l'état de l'art de l'année 2023.

Le chapitre II présente les travaux menés dans le cadre du premier contexte d'application et dont l'objectif initial est d'optimiser l'AC appliqués aux signaux des RSCFs. Les problématiques de reconstructions liées aux dimensions temporelle et spatiale des données sont abordées différemment au travers de trois méthodes de reconstructions de signaux hautement comprimés: temporelle, spatiale et spatio-temporelle. La méthode de reconstruction temporelle a fait l'objet d'une publication et d'une présentation à la 12ème IEEE (Institute of Electrical and Electronics Engineers) IDAACS "International Conference on Intelligent Data Acquisition and Advanced Computing Systems". La première partie du chapitre présente l'aspect méthodologique de la conception et du développement de ces méthodes. La seconde partie est du ressort de l'applicatif. L'expérimentation de ces méthodes a permis de démontrer leur efficacité dans le contexte de l'état de l'art et dans la réalisation des objectifs de la thèse. Les résultats de nos applications sont largement étudiés dans cette deuxième partie. La conception et la réalisation d'une Interface de Programmation d'Application (IPA), ou Application Programming Interface (API), dédiée à l'implémentation rapide et facile des méthodes mises au point est également introduite dans ce chapitre.

Le chapitre III traite entièrement de la littérature et des travaux de thèse qui concernent la simulation de structures colloïdales. Dans un premier temps, il présente l'institut de recherche sur les céramiques IRCER et la nature de notre collaboration. Il aborde les fondamentaux de physique exclusivement dédiés à la compréhension du travail d'identification paramétrique de l'application et aux problématiques de procédés céramiques de l'IRCER. Dans un second temps, la partie technique, qui inclut la conception et la réalisation de modèles d'AP et les expérimentations sur des jeux de données de colloïdes de l'IRCER, est développée dans sa complexité, ses problématiques et son aboutissement.



Etat de l'art

Sommaire

I.1	Connaissances fondamentales	24
I.1.1	Les réseaux de capteurs sans fil	24
I.1.1.1	Définition d'un réseau de capteurs sans fil	24
I.1.1.2	Enjeux, problématiques et applications	25
I.1.1.3	Caractérisation des signaux d'un réseau de capteurs sans fil	26
I.1.2	L'acquisition comprimée	27
I.1.2.1	Théorie fondatrice	27
I.1.2.2	Acquisition comprimée distribuée	28
I.1.2.3	Dédiée aux réseaux de capteurs sans fil	29
I.1.2.4	Quelques optimisations de la théorie	32
I.1.3	L'apprentissage profond	34
I.1.3.1	Un peu d'histoire	34
I.1.3.2	Les réseaux de neurones artificiels	34
I.1.3.3	Processus d'apprentissage	35
I.1.3.4	La descente de gradient	37

I.1.4	Les classes de réseaux de neurones	39
I.1.4.1	A propagation avant	39
I.1.4.2	Récurrent	39
I.1.4.3	Graphique	41
I.1.4.4	Convolutif	41
I.1.4.5	Les hyperparamètres	43
I.2	Apprentissage profond en lien avec les objectifs de la thèse	45
I.2.1	Des modèles de réseaux de neurones	45
I.2.1.1	Le perceptron multicouche	45
I.2.1.2	L’auto-encodeur	45
I.2.1.3	Le réseau antagoniste génératif	46
I.2.1.4	Le Transformeur	48
I.2.1.5	Le modèle de diffusion	49
I.2.2	Prévision des données	50
I.2.2.1	Introduction	50
I.2.2.2	A base de réseaux de neurones récurrents	50
I.2.2.3	Diverses solutions	51
I.2.3	Imputation de données manquantes	53
I.2.3.1	Introduction	53
I.2.3.2	Aux séries temporelles multivariées	53
I.2.3.3	Aux données raster	56

Ce chapitre regroupe les informations nécessaires à la compréhension des travaux de thèse. Il se concentre particulièrement sur le sujet du premier contexte d’application des RCSFs. L’étude qui y est menée sert à positionner notre approche et le développement des méthodes de la thèse au niveau de l’état de l’art de 2023 en matière d’AP.

La physique des particules ainsi que des références complémentaires d’AP spécifiques au second champ d’application des suspensions colloïdales se situent en première partie du chapitre III.

I.1 Connaissances fondamentales

I.1.1 Les réseaux de capteurs sans fil

I.1.1.1 Définition d’un réseau de capteurs sans fil

Un RCSF [1] ou Wireless Sensor Network (WSN) est un ensemble de nœuds de capteurs répartis dans l’espace et connectés les uns aux autres par un moyen de communication sans fil. Ils constituent un réseau. Chaque nœud est composé d’au moins un capteur et d’un dispositif de communication, le plus souvent un émetteur-récepteur radio. Il peut également intégrer de l’équipement embarquable tel qu’un microprocesseur, une mémoire ou une batterie d’alimentation. Les nœuds sont, soit des routeurs, soit des terminaux. Un terminal transmet des paquets de données mais n’en reçoit pas, contrairement au routeur qui peut relayer les messages provenant d’autres nœuds. Il permet, en l’occurrence, de concevoir plusieurs topologies de réseau différentes et d’étendre un réseau existant avec de nouveaux nœuds, voir Figure I.1. Chaque réseau possède une station de base chargée de démarrer le réseau et d’exécuter les protocoles de communication.

Un capteur est une unité d’acquisition qui capture un phénomène physique et le transforme en une grandeur interprétable. Il est au moins composé d’un dispositif de mesure conçu pour quantifier le phénomène observé et d’un transducteur effectuant la conversion de son énergie en un signal électrique. Il peut également disposer d’un système électronique de conversion qui permet d’amplifier, de filtrer et digitaliser le signal électrique.

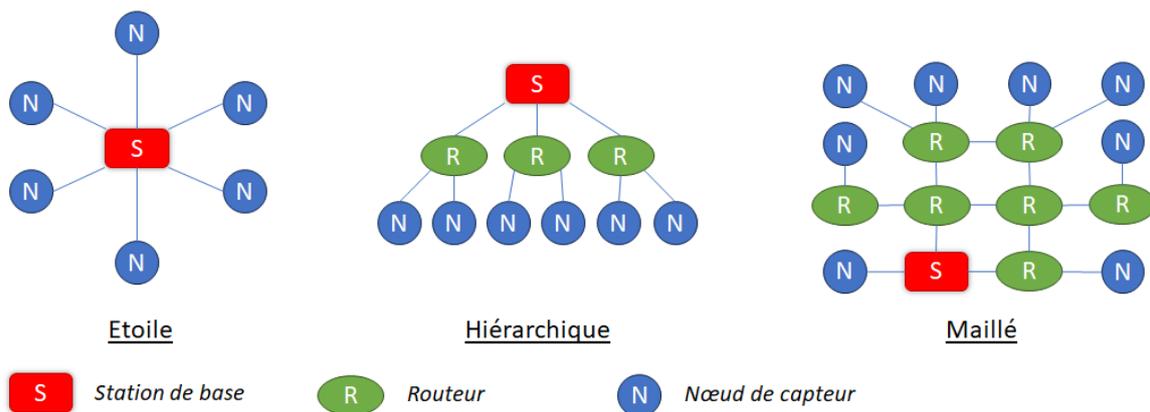


Figure I.1: Topologies des réseaux de capteurs sans fil

I.1.1.2 Enjeux, problématiques et applications

Les déploiements de RCSFs ont proliféré ces vingt dernières années [2] dans le but de surveiller des zones (applications militaires, transports, industries...) ou d'observer des phénomènes naturels (applications dans le domaine de la santé, écologie, études environnementales ou climatiques...). Les enjeux de leurs applications peuvent être d'intérêts économique, écologique, de santé ou de sécurité. La fiabilité des informations qu'ils mesurent est cruciale pour une bonne interprétation, l'analyse et la prise de décision.

Selon les applications, des problématiques de résistance des équipements, d'imperméabilité, de mobilité des nœuds, de communication longue distance, de système dynamique, de sécurité des données sensibles peuvent survenir. La problématique commune à toutes les applications reste l'autonomie. Les nœuds de capteurs doivent être les plus autonomes possible pour ne pas conduire à des pertes ou des altérations des mesures. Cela survient lors de dysfonctionnements techniques tels qu'une défaillance de l'équipement, une instabilité de la communication réseau, une saturation des bandes de fréquences radioélectrique ou du stockage des bases de données. Par ailleurs, les nœuds fonctionnent le plus souvent sur batterie pour faciliter leur déploiement, ce qui pose un problème d'autonomie énergétique. Aujourd'hui, la littérature met l'accent sur la recherche de solutions parcimonieuses en demande énergétique qui garantissent un fonctionnement fiable et sans interruption de service tout en minimisant la consommation d'énergie.

Cette thèse n'a pas pour objectif d'étudier les solutions matérielles liées aux problématiques des RCSFs. Elle propose des solutions logicielles qui se positionnent au niveau

des couches supérieures (l’acquisition et le traitement des données).

I.1.1.3 Caractérisation des signaux d’un réseau de capteurs sans fil

Du point de vue de la physique, les mesures acquises par un capteur au cours d’un intervalle de temps forment ce que l’on appelle un signal qui donne une représentation discrète du phénomène physique observé. Les signaux qui proviennent simultanément, c’est à dire dans le même intervalle de temps, des noeuds de capteurs d’un même RCSF forment ce que l’on appelle un ensemble de signaux. Si des corrélations spatiales existent entre les phénomènes physiques observés par les noeuds de capteurs d’un RCSF alors ces signaux partagent des caractéristiques communes. Pour la suite de la thèse, on appelle un **signal complet** ou continu, un signal qui comprend un nombre de mesures échantillonnées selon un taux au moins équivalent à celui de Nyquist défini par le théorème d’échantillonnage de Nyquist-Shannon. C’est à dire que la fréquence d’échantillonnage est supérieure au double de la fréquence de Nyquist. Cette fréquence est la fréquence maximale que contient un signal qui décrit sans ambiguïté le phénomène physique observé. Une valeur issue d’un signal complet sera désignée sous l’appellation **valeur réelle**, sous entendu, valeur qu’on admet être fiable et représentative de la réalité puisqu’elle provient directement d’une mesure d’un noeud de capteurs.

Du point de vue des mathématiques, études statistiques et AA, un signal de RCSF est connu en tant que Série Temporelle (ST), c’est-à-dire, en tant que série de valeurs indexées par une échelle de temps. Un ensemble de signaux représente des données spatio-temporelles collectées depuis des positions (celles des noeuds de capteurs) **irrégulièrement réparties** dans un espace défini, dans la majorité des cas. Cette particularité place un ensemble de signaux dans la catégorie des données «**raster**» [3], terme réutilisé dans la thèse.

I.1.2 L’acquisition comprimée

I.1.2.1 Théorie fondatrice

La théorie de l’AC est apparue avec l’article fondateur de Donoho [4]. L’AC permet la reconstruction d’un signal compressé en le projetant dans une base dans laquelle le signal original est parcimonieusement représenté. Son principe élémentaire étant qu’un signal peut être représenté de façon plus parcimonieuse dans une base différente de la base canonique. On dit qu’un signal est k -parcimonieux dans une base de représentation lorsque k représente le nombre d’élément non nul au sein de cette base de représentation. Cette méthode de compression permet d’atteindre des taux d’échantillonnage bien inférieurs à celui requis par le théorème de Shannon-Nyquist tout en ayant très peu de perte d’information.

L’AC repose sur une modélisation simple des signaux temporels sous la forme d’une combinaison linéaire de la forme donnée par l’équation :

$$x(t) = \sum_{i=1}^n \alpha_i \psi_i(t) \quad (I.1)$$

Où les fonctions $\psi_i(t)$ constituent les vecteurs de base, α_i les coefficients de projection et n la dimension choisie de l’espace de représentation. En toute rigueur cette dimension devrait être infinie pour des signaux continus. Néanmoins en choisissant judicieusement la base de représentation pour la classe de signaux considérée, il sera possible de tronquer la valeur de n pour ne garder que les vecteurs de base les plus représentatifs, avec $k \ll n$. Des bases typiques sont celles de Fourier ou encore d’ondelettes.

L’AC trouve des applications [5] dans les domaines qui incluent, sans y être limité, des mathématiques, du traitement de son, d’image et de vidéo, des télécommunications, de l’ingénierie électronique, de l’imagerie médicale.

Dans un contexte de réseau de communication numérique, les signaux sont discrétisés et nous considérons à présent que n représente également le nombre d’échantillons du signal x dans une fenêtre temporelle fixée de façon que la période d’échantillonnage satisfasse le critère de Shannon. On notera alors Ψ la matrice dont les colonnes sont constituées par les vecteurs de base $\psi_i(t)$ de dimension n , et α le vecteur des coefficients de projection tels que : $x = \Psi\alpha$.

Notons à présent y , le signal réellement observé, de longueur m , soit le nombre

d’échantillons du signal x dans la même fenêtre temporelle. Nous pouvons décrire le processus d’observation par un transfert linéaire caractérisé par la matrice de détection $\Phi = \{\varphi_{i,j}\}$ où les échantillons y_i de y sont dérivés depuis les échantillons x_j de x tel que :

$$y_i = \sum_{j=1}^n \varphi_{i,j} x_j, \quad i = 1, \dots, m \quad (1.2)$$

I.1.2.2 Acquisition comprimée distribuée

Dans le cas d’un RCSF, nous étudions un ensemble de plusieurs signaux corrélés entre eux spatialement et par des phénomènes physiques. L’AC exploite les corrélations intrinsèques à un signal seul. La première contribution scientifique majeure à l’étude de système multi-signaux est l’article de Baron [6] qui met en place l’Acquisition Comprimée Distribuée (ACD), ou Distributed Compressive Sensing (DCS). L’ACD généralise le principe de base d’AC par lequel un signal peut être parcimonieux dans une certaine base de représentation à un ensemble de signaux qui pourrait être conjointement parcimonieux dans une certaine base de représentation. Alors chaque signal d’un même ensemble peut être décrit par la combinaison d’une composante commune z_C , présente dans tous les signaux, et d’une composante d’innovation z_l , unique pour chaque signal, tel que :

$$x_l = z_C + z_l, \quad l \in \{1, \dots, L\} \quad (1.3)$$

Où L est le nombre de signaux de l’ensemble. L’ACD modélise l’ensemble des signaux grâce à un Joint Sparsity Model (JSM) noté \mathcal{P} , l’ensemble de toutes les sous-matrices identité de dimension $n \times n$, pour toutes les valeurs possibles de k , tel que l’on puisse exprimer un signal x par cette alternative :

$$x = P\theta, \quad P \in \mathcal{P} \quad (1.4)$$

P est composé de k colonnes d’une matrice identité de dimension $n \times n$ et θ contient les éléments non nuls du signal alors $\theta \in \mathbb{R}^k$. On note alors X , la matrice dont les colonnes sont faites des signaux de l’ensemble de signaux conjointement parcimonieux, tel que :

$$\mathbf{X} = \sum_{l=1}^L \mathbf{P}_C \boldsymbol{\theta}_C + \mathbf{P}_l \boldsymbol{\theta}_l, \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_C & \mathbf{P}_1 & 0 & \cdots & 0 \\ \vdots & 0 & \mathbf{P}_2 & 0 & \vdots \\ \vdots & \vdots & 0 & \ddots & 0 \\ \mathbf{P}_C & 0 & \cdots & 0 & \mathbf{P}_L \end{bmatrix} \quad (1.5)$$

En partant de la théorie de l’AC, l’ACD propose trois modèles de parcimonie qui exploitent, à la fois, les corrélations internes à un signal, et les corrélations qui existent entre les signaux. Son modèle *JSM-1* est spécifiquement adapté pour les applications aux RCSFs. Ce modèle suppose que les composantes z_C et z_l présentées dans l’équation 1.3 sont toutes les deux parcimonieuses. L’étude de l’ACD apporte également la preuve que l’utilisation de la norme l_1 dans ce contexte est le meilleur choix pour reconstruire les signaux compressés selon le modèle *JSM-1*. Les résultats obtenus par ACD surpassent en exactitude de reconstruction ceux obtenus par AC.

1.1.2.3 Dédicée aux réseaux de capteurs sans fil

L’AC et l’ACD sont largement utilisés dans l’IdO [7], [8], particulièrement avec les RCSFs, pour diminuer considérablement le nombre d’acquisition au niveau des nœuds. Cela permet de réduire drastiquement la consommation des batteries, les coûts de communication, l’agrégation de données sur le réseau et le volume stocké en base de données. La thèse de M. Itoua-Engoti [9] démontre notamment l’efficacité de l’ACD appliquée à un RCSF. La méthode est mise œuvre sur des mesures de température réalisées par une trentaine de capteurs répartis dans un bâtiment à 3 étages. Les corrélations spatiales existantes entre ces capteurs ont permis à l’ACD de diminuer les erreurs de reconstruction des signaux reconstruits comparativement à la reconstruction unique d’un signal.

Pour cette application, c’est plus précisément la méthode Acquisition Comprimée de Kronecker (ACK) [10], ou Kronecker Compressive Sensing (KCS), qui a été choisie. Plus récente que l’ACD, cette méthode exploite les prédispositions des produits de Kronecker à offrir des bases de représentations parcimonieuses pour un ensemble multi-signaux et des matrices de détection adaptées à de l’AC. Dans leur article [10], Duarte et Baraniuk mentionnent la parenté de l’ACK avec l’ACD [6] en mettant en lumière ses limites en conditions réelles. En effet, l’ACD considère dans sa théorie des signaux parfaits, sans perturbation ni bruit et ne s’adapte pas correctement à tous types de structures multi-signaux.

L’ACK considère un ensemble de signaux comme étant un seul signal \bar{x} à trois dimensions résultant du produit de Kronecker de l’ensemble des signaux. Ensuite, son étude établit la preuve que cette notion peut être étendue à la théorie de l’AC. Alors la base de représentation dans laquelle l’ensemble des signaux est conjointement parcimonieux est la résultante du produit de Kronecker des bases de parcimonie de chaque signal. Le principe est valable aussi pour la matrice de détection. Ce qui permet d’exprimer :

$$\bar{x}(t) = \sum_{i=1}^n \Theta_i \bar{\psi}_i(t), \quad \bar{\psi}_i = \bar{\psi}_1 \otimes \dots \otimes \bar{\psi}_L \quad (1.6)$$

Où Θ sont les coefficients de projection pour le signal \bar{x} . Alors la reconstruction de \bar{x} permet la reconstruction de tous les signaux de l’ensemble.

L’approche spatio-temporelle développée par M. Itoua-Engoti [9] consiste à utiliser, comme base de décomposition des signaux, une combinaison d’un vecteur fréquentiel obtenu par une Transformée en Cosinus Discrète (TCD) et d’un vecteur spatial obtenu par analyse préalable des inter-corrélations entre les différents noeuds de capteurs répartis dans l’espace. Pour ce faire, il faut calculer la matrice d’inter-corrélations suivante:

$$\Sigma = \frac{1}{K} \sum_{k=1}^K (\mathbf{x}_k - \bar{\mathbf{x}}) \cdot (\mathbf{x}_k - \bar{\mathbf{x}})^T \quad (1.7)$$

Où K est le nombre de capteurs, $\mathbf{x} = \{x_1 \dots x_K\}$ est le vecteur des mesures de tous les capteurs à un instant t et $\bar{\mathbf{x}} = \{\bar{x}_1 \dots \bar{x}_K\}$ est le vecteur des moyennes temporelles des différents capteurs. La base de décomposition est alors élaborée à partir d’une décomposition en composantes principales. La matrice de projection est la matrice de Décomposition en Valeurs Singulières (DVS) U_N , où N est le nombre d’échantillons temporels. La base spatio-temporelle de décomposition des signaux est le produit de Kronecker entre la matrice transposée de U_N et la matrice obtenue par TCD pour le domaine temporel. Soit:

$$\mathbf{W} = U_N^T \otimes TCD_K \quad (1.8)$$

Les mesures réalisées par un groupe de 13 capteurs sont représentées à la Figure I.2 suivant les deux axes temporel et spatial. Le spectre spatio-temporel correspondant à ces mesures et obtenu par la transformation \mathbf{W} est donné à la Figure I.3. On retrouve

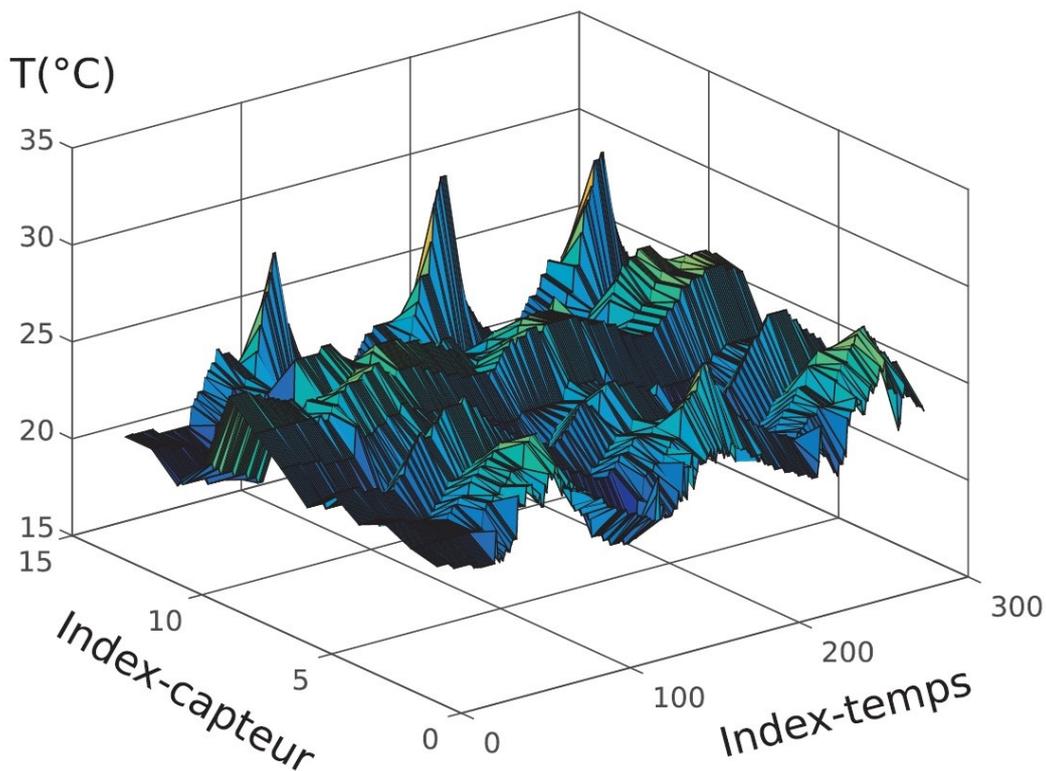


Figure I.2: Ensemble des mesures des capteurs placés au 3ème étage de l’IUT GEII de Brive-la-Gaillarde entre le 20/09/2016 et le 21/09/2016 à 23h59 (figure issue de [9])

N îlots de fréquences spatio-temporelles qui correspondent aux N valeurs singulières de la transformation DVS. Le nombre de composantes dont la valeur est supérieure à 1% de la valeur maximum est de 323 dans ce cas. Donc, la transformation spatio-temporelle obtenue par le produit de Kronecker des matrices de transformation spatiale et temporelle définit bien une représentation parcimonieuse du signal spatio-temporel et permet donc l’utilisation d’une méthode d’AC pour restituer le signal complet.

La méthode développée au cours de ce travail a donné de bons résultats (certains sont présentés dans le chapitre II), elle est cependant limitée par deux écueils:

- Contrairement à la décomposition de Fourier, la matrice de projection spatiale n’est pas connue a priori. Son élaboration nécessite donc un apprentissage qui permet de déterminer la valeur moyenne des signaux et leur corrélation. Cet apprentissage doit être réalisé avec des signaux non compressés.
- La taille de la matrice W croît très rapidement avec le nombre de capteurs, entraînant des temps de calcul qui peuvent devenir très rapidement prohibitifs

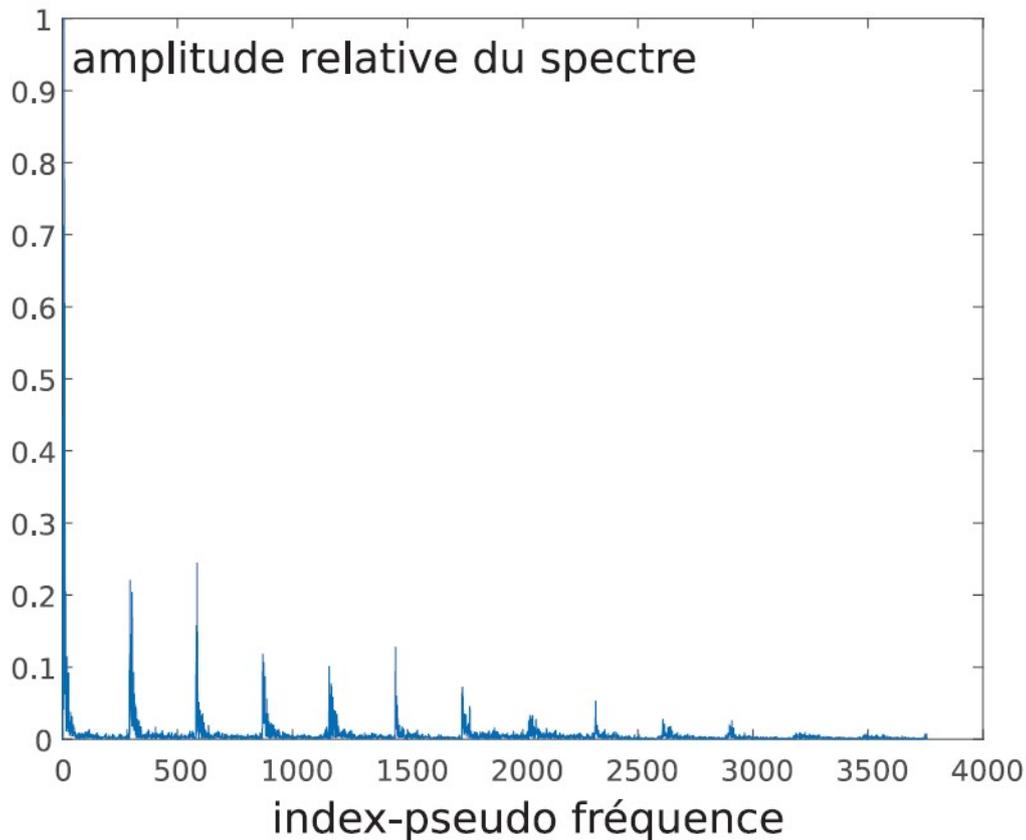


Figure I.3: Spectre du vecteur champ spatio-temporel obtenu par la transformation W (figure issue de [9])

comme on pourra le constater au chapitre II où nous avons comparé les résultats obtenus par cette méthode et celle développée au cours de notre travail.

I.1.2.4 Quelques optimisations de la théorie

Le problème de la restauration du signal x à partir du signal y , sachant que $m < n$, est un problème mal posé au sens de Hadamard. Ainsi, la recherche de solution pour trouver les coefficients de projection α a été le sujet d’énormément de travaux [11] depuis l’émergence de l’AC qui exploite la parcimonie des signaux dans la base Ψ et un choix judicieux de la matrice de mesure pour obtenir une excellente estimation de α . Ils permettent aujourd’hui d’adapter le calcul des coefficients de projection à un taux d’échantillonnage, à la nature du signal et aux performances visées. Les méthodes les plus populaires sont : les algorithmes gloutons à base de Matching Pursuit (MP), les algorithmes de l_p -minimisation (l_0, l_1, l_2), les méthodes Bayésiennes [11].

Parmi les travaux qui optimisent l’AC à différents niveaux, l’AP a fortement impacté l’AC dans les domaines du traitement de l’image et de la vidéo, aussi l’image à résonance magnétique [12]–[16]. Ce n’est pas le cas du domaine des RCSFs, on peut tout de même citer des méthodes à bases d’AP qui cherchent à trouver une représentation plus parcimonieuse du signal [17], [18] ou à améliorer la qualité de reconstruction [19]–[22]. Par exemple, l’emploi d’un RN peut remplacer une détection aléatoire pour obtenir de meilleures représentations parcimonieuses d’un signal. Sur la base d’un apprentissage de la nature des signaux à reconstruire, il estime pour chaque position d’une matrice détection les probabilités d’y trouver une valeur discrète significative. D’autre part, un RN peut aussi reconstruire les signaux compressés en remplaçant ainsi les algorithmes de MP et de minimisation.

I.1.3 L’apprentissage profond

I.1.3.1 Un peu d’histoire

On parle d’AP lorsqu’on parle d’apprentissage de représentations de données par la mise en œuvre d’un algorithme de rétropropagation [23] dans un réseau de neurones artificiels. Le principe de rétropropagation consiste à calculer une dérivée sur une courbe d’erreur en fonction d’un poids en vue de modifier la valeur du poids selon la pente observée pour minimiser l’erreur. La suite du chapitre permet de saisir l’application de ce principe dans l’AP. L’idée d’un RN artificiel remonte en 1943 [24], elle est issue de la collaboration d’un neurophysiologiste et d’un mathématicien, respectivement Warren McCulloch et Walter Pitts. Leurs neurones réalisent un circuit d’opérations logiques. L’idée est aussi pensée et expérimentée par Frank Rosenblatt en 1957 avec sa réalisation d’un perceptron [25]. Son modèle est plus complexe : il intègre des poids variables dans la formule d’un neurone artificiel pour permettre l’évaluation de l’importance des données qui transitent en son entrée.

L’AP ne se popularise qu’autour des années 2010 grâce à l’arrivée de nouveaux outils technologiques, notamment de puissants processeurs graphiques pouvant réaliser plus de mille milliards d’opérations par seconde. Ils permettent d’implémenter des RNs artificiels et d’exécuter leurs algorithmes de fonctionnement sans occasionner de dépassement de mémoire et sans atteindre des temps d’exécution démesurés. On rappelle aussi que cette période coïncide avec l’explosion de l’IdO et l’utilisation d’Internet qui résulte en une quantité massive de données stockée dans les centres dédiés et qui ne cessent de s’accroître.

L’AP n’a pas pu trouver d’applications entre les années 1960 et 2010 et donc d’investissement pour la recherche à cause d’un manque de données et de matériel informatique assez puissant pour les rendre efficaces. Durant ces années, des chercheurs comme Yann Lecun, Yoshua Bengio et Geoffrey Hinton ont poursuivi les travaux de Frank Rosenblatt et David Rumelhart et fondent le Deep Learning [26] dans la perspective d’ouvrir la voie à des applications innovantes de l’AP dans tous les domaines de la recherche capable de produire et d’enregistrer de la donnée.

I.1.3.2 Les réseaux de neurones artificiels

Un neurone artificiel, bien qu’inspiré de la neurobiologie, ne fonctionne en rien comme un neurone biologique. Sa forme la plus simple d’aujourd’hui est un perceptron, Figure I.4,

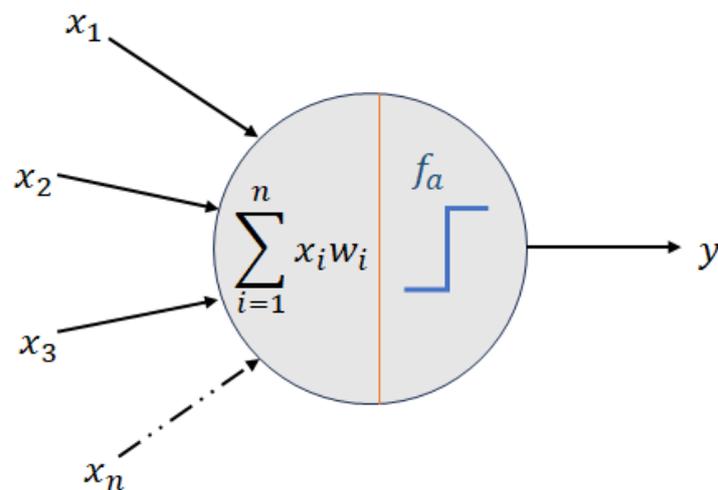


Figure I.4: Neurone artificiel - Perceptron

aussi appelé unité logique à seuil.

Il intègre un nombre n de données en son entrée qu’il pondère par ses poids internes w et en fait la somme. Le résultat de cette opération est filtré par une fonction d’activation f_a pour donner sa sortie y .

Les RNs sont constitués de cellules dites neuronales, donc de neurones artificiels, structurées en des couches successives. La notion de profondeur est apportée par le nombre de couches qui contribuent à modéliser un type de donnée. Elles sont assimilées par le réseau à partir d’un point d’entrée, transitent dans les couches neuronales qui produisent une sortie correspondante.

I.1.3.3 Processus d’apprentissage

Un RN est conçu pour être soumis à un apprentissage qui lui permet de produire, par lui-même, une sortie attendue en paramétrant ses poids internes aux valeurs qui permettent de faire correctement correspondre une donnée d’entrée avec une donnée de sortie. Ce processus d’ajustement des poids est réalisé par rétropropagation de la dernière à la première couche en commençant par constater l’erreur entre la sortie attendue et la sortie produite par le biais d’une fonction de perte. A travers plusieurs cycles d’injection de données, que l’on appelle un entraînement, l’objectif de l’algorithme de rétropropagation est de trouver l’ajustement optimal des poids qui conduit à réduire l’erreur calculée par la fonction de perte. La première application

notable de cet algorithme dans un contexte d’AP est celle de Yann LeCun [27].

L’algorithme qui choisit les variations de valeurs des poids qui sont appliquées lors d’une rétropropagation est appelé un optimiseur. Comme illustré Figure I.5, il est celui qui interprète l’erreur entre les sorties du RN et les données cibles.

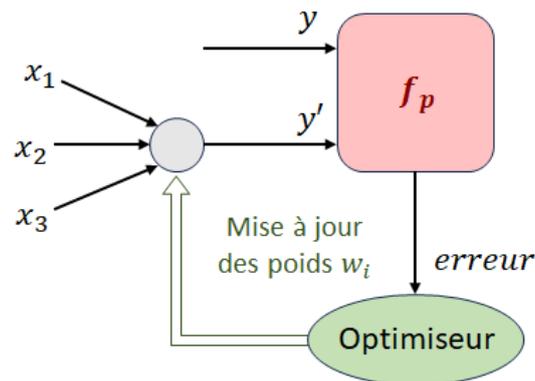


Figure I.5: Ajustement des poids neuronaux par rétropropagation

Les phases d’entraînements sont soumises à deux phénomènes indésirables : le surajustement et le sous-ajustement. Un jeu de donnée d’entraînement sert à apprendre au RN à modéliser et généraliser un problème pour extraire depuis les données les règles qui induisent une réponse particulière à une donnée spécifique. Or, dans la mise en œuvre d’un RN entraîné, les données qu’il traite ne sont majoritairement pas incluses dans le jeu d’entraînement. Il existe donc une frontière durant l’apprentissage entre le moment où les poids internes d’un RN ne sont pas assez ajustés pour son utilisation et le moment où ils sont trop ajustés au jeu de données d’entraînement, comme illustré sur la Figure I.6.

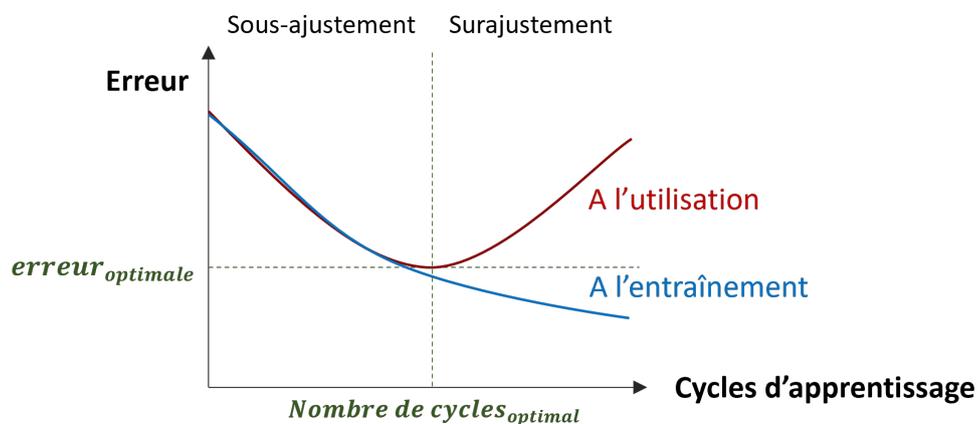


Figure I.6: Phénomènes d’ajustements au jeu de données d’entraînement

Il existe de nombreuses techniques pour contrôler ces phénomènes et trouver l’ajustement optimal des neurones [28], [29].

On distingue quatre catégories d’apprentissage différents en fonction du moyen utilisé pour entraîner un RN :

- supervisé, il s’agit exactement du processus d’apprentissage précédemment décrit. L’apprentissage supervisé met en relation des données d’entrées avec des données cibles, connues pendant l’entraînement.
- non supervisé, il s’agit de transformer la donnée sans cibler de résultat pour en changer les dimensions ou la base de représentation. C’est très souvent une étape préalable à l’apprentissage supervisé.
- auto-supervisé, il s’agit d’un apprentissage supervisé dont les données cibles ne sont pas définies par un humain. Le cas typique est celui de données cibles générées par un RN déjà entraîné, utilisé dans le processus d’apprentissage d’un autre.
- par renforcement, il s’agit d’un apprentissage très particulier qui consiste à entraîner un RN à générer des actions pour maximiser une récompense. Autrement dit il n’y a pas de données cibles sinon un score de réussite et il n’y a pas de données d’entrée sinon l’état du système actuel. La génération de données, de résultats, se fait simultanément à l’entraînement. Pour plus d’informations : [30].

Les travaux de cette thèse se situent dans le cas d’un apprentissage supervisé, qui peut inclure des étapes d’apprentissage non supervisé et auto-supervisé sans pouvoir s’y suffire.

I.1.3.4 La descente de gradient

Comme introduit dans la section précédente: dans l’espace des possibilités des valeurs des poids internes à un RN, il existe autant de résultats de la fonction de perte au nombre d’échantillon de données du jeu d’entraînement. Chacun de ces résultats est une courbe qui mesure l’erreur en fonction des valeurs des poids du RN. Un gradient se confond avec une dérivée sur une de ces courbes, comme illustré sur la Figure I.7.

C’est un tenseur qui a pour composantes les dérivées partielles de la fonction d’erreur par rapport aux poids d’un neurone. La descente de gradient consiste à faire varier les poids selon les gradients dans les directions qui réduisent l’erreur de la fonction

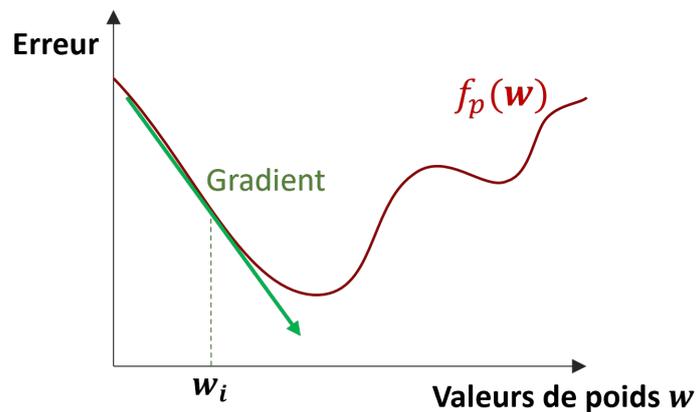


Figure I.7: Représentation d’un gradient sur une courbe d’erreur pour un échantillon de donnée

de perte. Un algorithme de descente de gradient sert à paramétrer les variations. Par exemple, l’algorithme de descente de gradient stochastique calcule les gradients pour des échantillons de données pris au hasard et recherche analytiquement les erreurs les plus basses pour lesquelles les dérivées composant les gradients s’annulent. Il existe d’autres variantes de cet algorithme [31] qui présente tous des avantages en terme de vitesse d’entraînement, de taux de convergence, de coûts et de performances à trouver le minimum d’erreur global. Il est mis en œuvre par l’optimiseur pendant l’apprentissage.

Les problèmes que rencontrent le plus fréquemment la descente de gradient sont la disparition et l’explosion des gradients. La méthode d’apprentissage par rétropropagation procède à la mise à jour des gradients de la dernière à la première couche du RN. Cela implique que les premières variations qui sont effectuées sur la dernière couche du réseau perturbent l’interprétation de l’erreur de la fonction de perte qui provient de la configuration initiale. Les gradients deviennent alors de plus en plus faibles jusqu’à, parfois, atteindre des valeurs nulles pour les premières couches du RN: c’est la disparition des gradients. Dans de rares cas, souvent lorsque la structure et l’hyperparamétrage du RN ne sont pas adaptés au problème qu’il doit résoudre, les courbes d’erreur sont trop irrégulières et peu généralisable alors le phénomène inverse se produit et les gradients deviennent de plus en plus grands: c’est l’explosion des gradients. Lorsque ces phénomènes indésirables se produisent l’apprentissage se fige.

I.1.4 Les classes de réseaux de neurones

I.1.4.1 A propagation avant

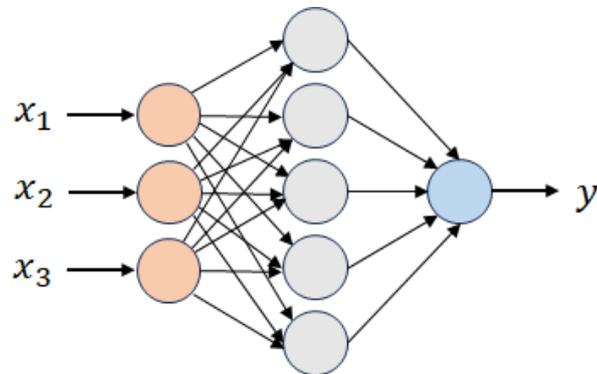


Figure I.8: Réseau de neurones à propagation avant

Un Réseau de Neurones à Propagation Avant (RNPA), ou Feed-forward Neural Network (FNN) est la classe la plus simple des RNs. Cette mention désigne tout RN étant acyclique, c’est à dire, tous les réseaux qui propagent les données à travers les neurones depuis l’entrée du réseau vers sa sortie et toujours vers l’avant, comme illustré Figure I.8. Ils sont opposés par nature aux réseaux de neurones récurrents. Le plus connu de cette classe est le perceptron.

I.1.4.2 Récurrent

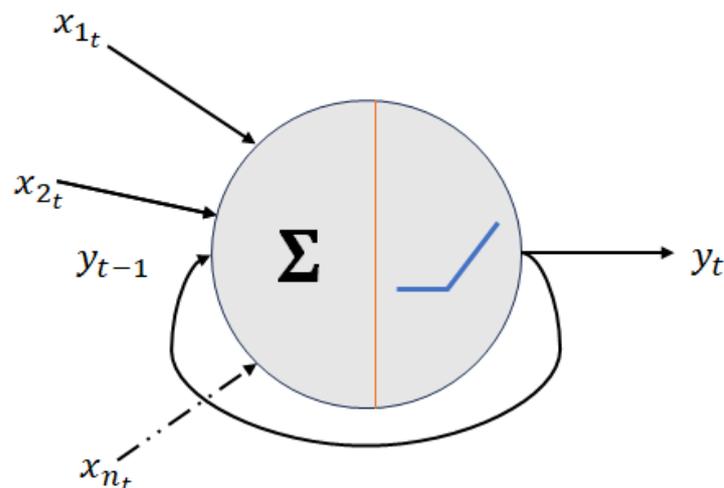


Figure I.9: Neurone artificiel récurrent

Les Réseaux de Neurones Récurrents (RNRs), ou Recurrent Neural Networks (RNNs), sont les premiers RNs spécialement conçus pour un type de donnée. Ils sont composés

de cellules neuronales dites récurrentes car elles connectent leur sortie à leur entrée, comme illustré Figure I.9. Les applications d’AP qui traitent de données temporelles ou séquentielles, notamment dans les domaines du traitement du langage et du traitement des signaux les utilisent largement.

Le RNR est victime d’un problème de disparition de gradients qui se produit dans la boucle récurrente. La profondeur du réseau amplifie exponentiellement la disparition de gradients dans l’évolution temporelle de l’erreur rétropropagée. Cela limite fortement les applications des RNR. En 1997, la cellule Long Short-Term Memory (LSTM) [32] est conçue par Hochreiter et Schmidhuber pour remédier spécifiquement à ce problème. Cette cellule de neurone artificiel récurrent, illustrée Figure I.10, diminue considérablement la disparition de gradients par répartition de ces derniers sur des états parallèles qui réduisent l’appauvrissement graduel de l’erreur temporellement rétropropagée. Elle améliore également les performances des RNRs avec une mémoire interne mieux structurée. Cette cellule complexifie le principe de récurrence en répartissant les tâches liées à la mémoire à travers ce qui est appelé des portes. Ces portes réalisent des opérations sur les entrées et sorties de la cellule. Cette dernière comporte une sortie supplémentaire appelé l’état de la cellule. Cet état conserve une mémoire à long terme des données.

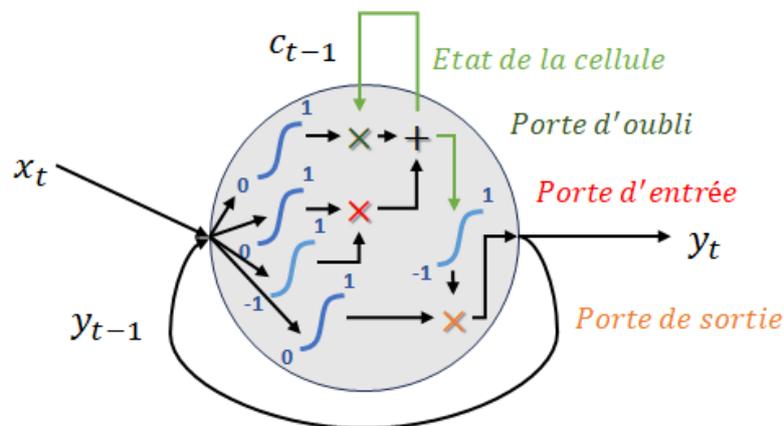


Figure I.10: Cellule récurrente Long Short-Term Memory

La cellule possède quatre matrices de poids, une pour chacune de ses portes et la dernière pour son état. Les rôles de chaque porte sont les suivants : la porte d’entrée extrait l’information de la donnée courante et précédente, la porte d’oubli met à jour l’état de la cellule, la porte de sortie détermine la donnée sortante en fonction de l’état de la cellule, l’état précédent et l’état courant. Contrairement à une cellule récurrente simple, la cellule LSTM peut conserver durablement une information importante en

mémoire pour s’en servir lorsqu’elle sera utile.

La cellule Gated Recurrent Unit (GRU) [33] est une alternative à la cellule LSTM avec moins de paramètres internes puisqu’elle ne comporte que deux portes et une seule sortie. Elle est plus rapide tout en parvenant à opérer sensiblement les mêmes performances que la cellule LSTM [34]. Elle a suscité un fort engouement dès son apparition en 2014 et demeure à l’état de l’art. Aucune des deux cellules ne se distingue réellement, selon les applications et les jeux de données, l’une est légèrement plus performante que l’autre ou vice versa.

I.1.4.3 Graphique

Les Réseaux de Neurones Graphiques (RNGs), Graph Neural Networks (GNNs), sont les derniers à avoir été inventés [35], [36] en 2005 pour répondre à un type de structure de données de plus en plus répandu dans le commerce et les services numériques comme les réseaux sociaux, les sites de ventes en lignes, les sites de recommandations. Ils sont conçus pour traiter des données d’entrée structurées en graphe, c’est à dire, sans dimensions fixes mais formatées en nœuds et en arêtes. Chaque nœud contient les caractéristiques des données et est relié à d’autres nœuds par des arêtes qui peuvent être orientées ou non, labelisées ou non. Ils permettent à un RN d’apprendre un nouveau type de corrélation entre les données qui n’est ni temporel, ni spatial mais social.

Ces réseaux sont utiles pour observer comment des informations circulent à travers un graphe ou pour faire correspondre un profil avec un produit ou un autre profil. On lui trouve également de plus en plus d’applications [37].

I.1.4.4 Convolutif

Les Réseaux de Neurones Convolutifs (RNCs), ou Convolutional Neural Networks (CNNs), sont une évolution du Neocognitron de Fukushima [38]. Ils sont les réseaux de neurones les plus utilisés et les plus étudiés [39]. Ils trouvent des applications dans tous les domaines qui touchent à l’AP. Cela tient au fait que leur structure interne est une version plus élaboré des RNPA qui leur permet une large compréhension de l’ensemble des données et d’éviter les problèmes de surajustement et sous-ajustement des gradients en plus d’être plus performants. Leur conception initiale est dédiée aux données en deux dimensions. Ce qui les rend particulièrement adaptés pour traiter des images. Leurs performances ont incité les chercheurs à les décliner pour des

données en une ou trois dimensions. Leur popularité est également à l’origine de combinaison avec les RNRs, comme le modèle ConvLSTM [40], et les RNGs, comme le modèle GraphCNN [41].

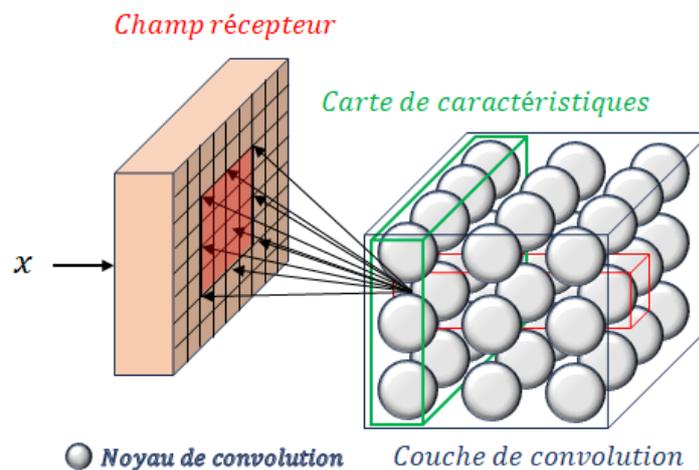


Figure I.11: Couche de convolution

La convolution est une opération mathématique proche des transformées de Fourier et de Laplace qui résulte en l’intégrale du produit de deux fonctions dont une glissante sur l’axe des abscisses. Les couches de convolution d’un RNC réalisent sensiblement la même opération, on parlera plutôt de corrélations croisées. Dans une couche de convolution les neurones sont connectés à une portion particulière de la donnée d’entrée que l’on appelle un champ récepteur, voir Figure I.11. Les couches sont constituées de strates profondes que l’on appelle cartes de caractéristiques qui sont elles-mêmes constituées de noyaux de convolution. Ainsi la compréhension globale de la donnée est une compilation de représentations locales et de représentations de caractéristiques. Le nombre de noyaux de convolution correspond au nombre de neurones d’une couche de convolution. La couche de convolution est dimensionnée à partir de trois paramètres : la profondeur, le pas et la marge. La profondeur représente le nombre de cartes de caractéristiques, c’est à dire le nombre de noyaux de convolution associés à un champ récepteur. Le pas définit la distance entre chaque champ récepteur, plus il est petit plus le nombre de convolution, donc la taille de la sortie de la couche, est élevé. La marge est la valeur qui comble les valeurs manquantes aux bordures de la donnée dans le champ récepteur. La taille du champ récepteur et du noyau de convolution sont identiques et paramétrables, on parle aussi de la taille du filtre en considérant que le noyau de convolution applique une sorte de filtre au champ récepteur. Les poids internes d’un noyau de convolution sont les paramètres entraînaibles d’un RNC.

I.1.4.5 Les hyperparamètres

L’implémentation d’un RN implique de déterminer des hyperparamètres. Les hyperparamètres désignent toutes les variables d’apprentissage qui ne peuvent pas être ajustées pendant l’apprentissage, autrement dit, ils ne sont pas entraînaibles. Un hyperparamètre peut participer à définir la structure interne d’un RN ou à contrôler son entraînement. Ils conditionnent l’IA, leurs rôles sont cruciaux et peuvent à eux seuls conduire à l’échec total d’un entraînement, d’une application, d’une méthode, d’un projet d’AP, s’ils sont mal ajustés aux objectifs. C’est pourquoi, toute publication scientifique relative à l’AP publie également ses hyperparamètres.

Certains hyperparamètres sont propres à une classe de RN, d’autres sont communs à tous les RNs. On retrouve notamment sans s’y limiter :

- les fonctions d’activation, ce sont des fonctions mathématiques qui filtrent les sorties des neurones pour entraîner une réponse. Quelques exemples de fonctions sont illustrés Figure I.12.

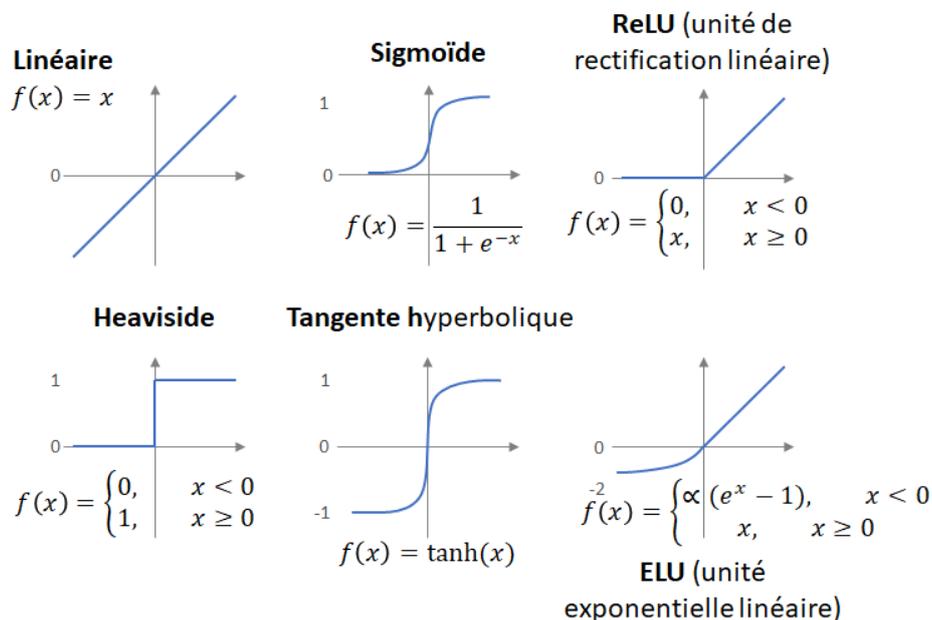


Figure I.12: Les fonctions d’activations les plus communes

- les fonctions de pertes, ce sont des fonctions qui mesurent le score, la perte, ou l’erreur entre les sorties du RN et les données cibles. Elles sont programmées en fonction de la nature de la donnée et surtout du problème à résoudre. Les plus connues sont les fonctions d’entropie croisées pour les problèmes de classification

et les fonctions d’erreur quadratique moyenne, Mean Squared Error (MSE), ou absolue, Mean Absolute Error (MAE), pour les problèmes de régression.

- les optimiseurs, ce sont des fonctions qui contrôlent la descente de gradients.
- le nombre d’époques, une époque désigne un cycle d’apprentissage. C’est donc le nombre de fois où les données d’entraînement transitent dans le RN pendant un entraînement.
- la taille d’un lot, cette taille désigne le nombre d’échantillons du lot de données d’entraînement qui transitent dans le RN avant une descente de gradients.
- le nombre de couches, il s’agit du nombre de couches de neurones.
- le nombre de neurones, il s’agit du nombre de neurones dans chaque couche.

Depuis la popularisation de l’AP dans la recherche, les modèles différents de RNs, d’optimiseurs, de fonctions d’activation et de perte sont devenus particulièrement variés, spécialisés et optimisés et cette tendance à la diversité et l’amélioration semble continuer [42]. De plus, il existe aussi des travaux qui visent à automatiser la recherche d’hyperparamètres optimaux pour la réussite d’un entraînement de RN [43].

I.2 Apprentissage profond en lien avec les objectifs de la thèse

I.2.1 Des modèles de réseaux de neurones

I.2.1.1 Le perceptron multicouche

Un perceptron multicouche est un modèle de RNPA. Il se différencie du perceptron [25] par :

- son nombre de couches, qui comprend une entrée, une sortie et au moins une couche cachée.
- des fonctions d’activations non linéaires.

Ce modèle est très populaire parce qu’il est très flexible et peut mettre en relation n’importe quelle donnée d’entrée avec n’importe quelle donnée de sortie à condition que la première peut induire la seconde. C’est le modèle universel de résolution de problèmes complexes par apprentissage. Ce modèle n’est dédié à aucune forme de donnée spécifique (temporelle, spatiale, graphique...) et peut être adapté pour des problèmes de régression ou de classification. Il est constitué d’un réseau de neurones qui fait circuler l’information vers l’avant dans lequel chaque cellule oriente la sortie par le biais de poids entraînés vers un résultat dont la forme est spécifiée lors de sa conception. Les hyperparamètres d’AP permettent de le structurer pour différentes applications. Ce qui fait sa force fait aussi ses limites. Lorsqu’un problème est correctement identifié, des modèles plus spécifiques à des données ou des domaines d’applications particuliers sont plus efficaces et performants.

Dans l’état de l’art de cette thèse, le perceptron multicouche apparaît dans certains travaux récents pour résoudre des problèmes de représentations complexes qui n’entrent pas dans les catégories de modèles existants.

I.2.1.2 L’auto-encodeur

Un AE est un modèle qui peut être de toutes les classes neuronales. Ce terme désigne le concept de deux entités co-dépendantes qui sont: un encodeur et un décodeur. L’entrée du décodeur est toujours la sortie de l’encodeur, comme le montre la Figure I.13. Un encodeur encode une donnée en une représentation de faible dimension, dite compressée. L’ensemble des représentations issues d’un jeu de données créent ce

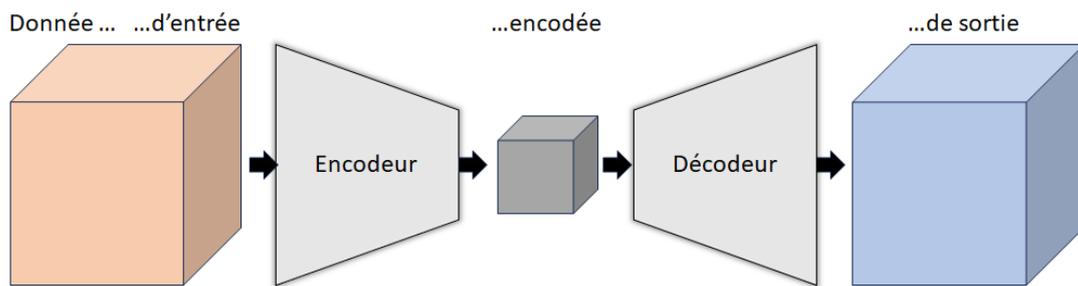


Figure I.13: Modèle simplifié d’auto-encodeur

que l’on appelle l’espace latent de l’AE. Donc, l’espace latent est un nouvel espace de représentation, généré par l’encodeur, de plus faible dimension que celui des données d’entraînement qui conserve et comprime les informations essentielles à la tâche qui est imputée au décodeur. Une représentation compressée contient, en quelques paramètres, la description d’une donnée au sein de l’espace latent qui peut être interprétée en une donnée de sortie attendue. Il existe de nombreuses variétés d’AE [44], le plus populaire étant l’AE variationnel.

L’AE est une méthode qui fait l’unanimité dans certains domaines, notamment la compression de données et le traitement d’image (débruitage, filtres, classification, détection d’objets, etc).

I.2.1.3 Le réseau antagoniste génératif

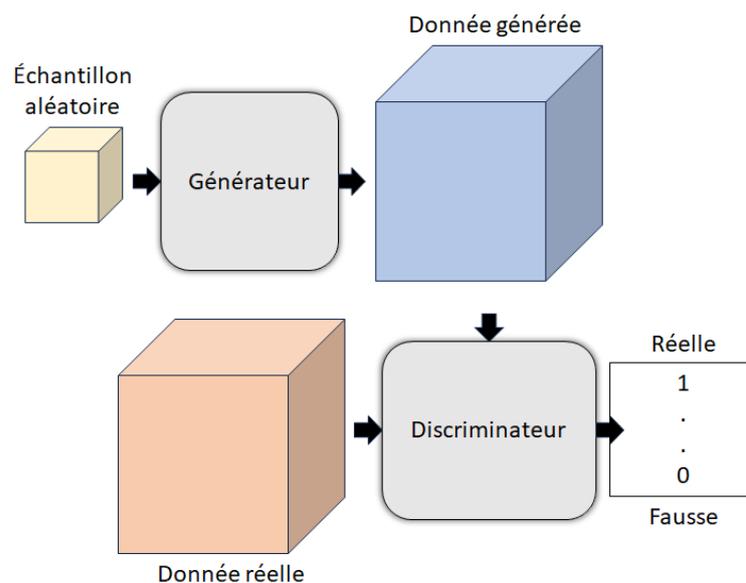


Figure I.14: Modèle simplifié de réseau antagoniste génératif

Un Réseau Antagoniste Génératif (RAG), Figure I.14, plus connu sous le nom de Generative Adversarial Network (GAN), est composé d’au moins deux modèles qui sont un modèle de générateur et un modèle de discriminateur. Ces derniers sont entraînés conjointement. Le discriminateur reçoit en entrée des échantillons de données réelles et de données générées par le modèle de générateur et apprend à les distinguer. Le générateur apprend à tromper le discriminateur. Le concept de RAG est unique car il permet de sortir du cadre du jeu de données d’entraînement pour générer ce qu’on appellera des données réalistes. Cette spécificité lui vaut un caractère «créatif» que les autres modèles n’ont pas et a conduit à son exploitation dans tous les domaines d’applications de l’AP [45]. Pour autant le RAG n’est pas une solution très flexible, sa créativité n’est pas utile et peut même être un désavantage pour des applications dans lesquels les données d’entraînement définissent précisément la structure du problème à résoudre. Par exemple, c’est le cas de données séquentielles avec une structure rigide pour des applications dans les domaines de traitement automatique du langage ou de traitement du signal.

C’est un modèle instable particulièrement difficile à entraîner. Il est victime de plusieurs phénomènes indésirables dont la disparition des gradients, plus particulièrement d’un mode de réduction, ou mode collapse, et d’un problème de divergence. La disparition des gradients se produit généralement chez le générateur lorsque que le discriminateur est trop performant. Il convient alors de modifier la structure du discriminateur pour ralentir son apprentissage en vue d’accélérer l’apprentissage conjoint des deux modèles. Le mode de réduction est un fonctionnement propre du générateur qui consiste à réduire l’espace de représentation des données à quelques échantillons suffisamment hétérogènes pour tromper le discriminateur. Le générateur effectue alors une rotation de génération dans cet ensemble et le discriminateur est piégé à discriminer chaque génération tour à tour dans une boucle sans fin. Le générateur devient alors limité et souvent inexploitable. La divergence est un problème très fréquent qui se produit lorsque les deux modèles ne parviennent pas à trouver un équilibre. Les générations sont rapidement discriminées car aberrantes et renouvelées car inédites, sans jamais converger vers les caractéristiques ciblées des données d’entraînement. Le générateur et le discriminateur doivent converger vers un équilibre dans lequel le discriminateur est quelque peu induit en erreur et où le générateur s’ajuste bien à l’espace de représentation des données. Beaucoup de travaux cherchent à résoudre ces problèmes pour faire du RAG une solution de RN universelle. Jusqu’à aujourd’hui, le RAG Wasserstein [46] est celui qui parvient à stabiliser au mieux son entraînement

et à minimiser l’apparition de ces phénomènes.

I.2.1.4 Le Transformeur

Le Transformeur s’apparente à un AE, car il est également composé d’un encodeur et d’un décodeur, qui utilise le mécanisme d’auto-attention [47]. Ce modèle, illustré sur la Figure I.15, représente une révolution très récente dans les domaines d’étude de données séquentielles, particulièrement dans le traitement automatique du langage naturel.

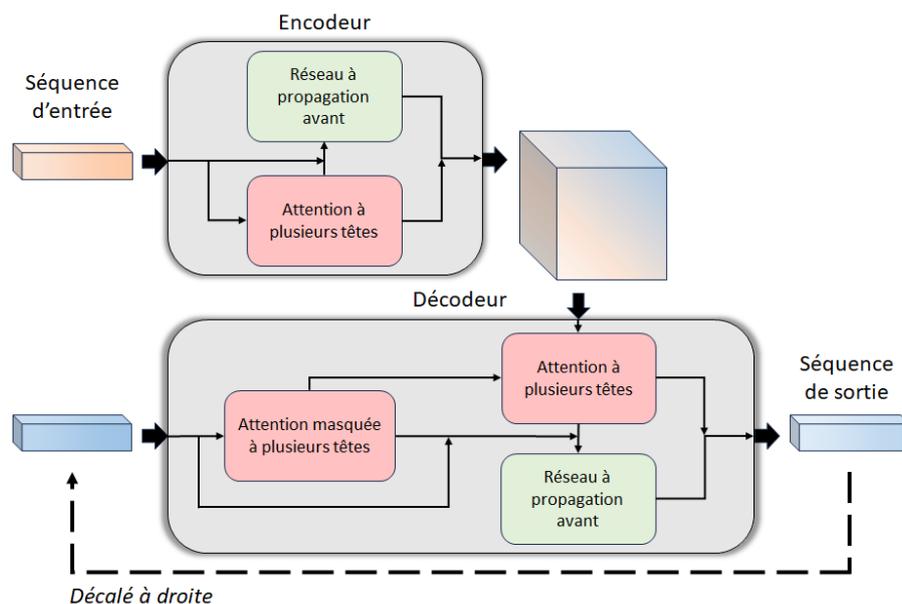


Figure I.15: Modélisation simplifiée du Transformeur

Le mécanisme d’auto-attention ou « self-attention » permet à chaque élément d’une séquence d’être traité par le réseau de neurones en considérant un vecteur dont les valeurs réfèrent aux autres éléments de la séquence pour pondérer leur influence et décrire son contexte. Autrement dit, il permet au RN de savoir à quelles informations précédentes et suivantes il devrait prêter attention dans son traitement courant. Ce mécanisme est dynamique, ces données contextuelles sont donc entraînaibles. L’attention à plusieurs têtes est un traitement en trois parties à trois entrées (requêtes, clefs, valeurs) différentes qui finissent concaténées en une sortie. Les entrées requêtes et clefs forment les paires d’éléments de la séquence d’entrée et les valeurs pondèrent leur relation.

Le Transformeur prétend à remplacer des modèles à base de RNR parce qu’il parvient à conserver en mémoire les relations entre chaque élément d’une séquence les uns avec les autres tout au long de l’entraînement, ce que les RNRs accomplissent difficile-

ment. Pour autant, il est aussi particulièrement complexe à concevoir efficacement et il nécessite beaucoup de données d’entraînement. Le Transformeur requiert plus de paramètres entraînaibles, donc d’utilisation de mémoire, que les autres modèles, ce qui ne le rend pas forcément populaire pour des applications qui ont des problématiques de coûts matériels, de rapidité et de faible consommation énergétique. C’est pourquoi on trouve relativement peu d’applications de ce modèle dans l’état-de-l’art, néanmoins chacune d’elles montre de remarquables performances dans leur domaine.

L’architecture flexible du Transformeur a permis de nombreuses déclinaisons pour réaliser des tâches spécifiques ou généraliser des problèmes complexes [48]. Les Transformeurs se sont imposés dans l’état de l’art du traitement du langage naturel. Les autres domaines afférents aux séries temporelles commencent aussi à substituer les Transformeurs aux RNRs pour divers traitements [49].

I.2.1.5 Le modèle de diffusion

Un modèle de diffusion est un modèle génératif de donnée basé sur le bruit. Ce modèle fonctionne également sur un principe d’AE en injectant du bruit dans la donnée d’entrée avec l’encodeur puis en débruitant vers la donnée de sortie avec le décodeur. Cette astuce permet de cibler un résultat tout en offrant la possibilité de restaurer de la donnée déjà bruitée. On distingue trois grandes méthodes pour réaliser la diffusion : des chaînes de Markov, des fonctions de score, des équations différentielles stochastiques. Ce modèle récent se popularise depuis quelques années pour le traitement d’image et du signal [50].

I.2.2 Prédiction des données

I.2.2.1 Introduction

La prédiction de STs par des méthodes d’AP est considérablement renseignée dans la littérature pour beaucoup d’applications dans les domaines du marketing, des télécommunications, de la finance, de l’énergie, de la climatologie, de la démographie, du transport, ou encore de l’écologie. Parce que les STs présentent des tendances, des motifs ou des cycles au cours du temps, les prévisions peuvent être exploitées pour de la surveillance, de l’analyse et de la prise de décisions. Il apparaît que tous les moyens sont bons pour réaliser cette tâche, la littérature contient des modèles convolutifs, des perceptrons multicouche, des RNRs mais également des transformeurs [51]. Tant de modèles qui ne trouvent pas de consensus mais sont surtout conçus spécialement pour leur domaine d’application, un formalisme de donnée et des objectifs de prédiction qui peuvent être aussi bien axés sur la précision, la longueur, une sortie probabiliste ou déterminée. Tant de travaux inspirants mais aucun qui permette d’affirmer qu’il est la solution.

Les travaux relatifs à la prédiction de STs et spatio-temporelles par AP sont connexes à cette thèse tant qu’une reconstruction est possible en prévoyant les données manquantes de cette façon. La différence majeure entre la prédiction et la reconstruction tient au fait que la prédiction s’opère à partir d’un jeu de données antérieur sur l’axe du temps tandis que le but de la reconstruction est d’estimer les données manquantes au sein d’une séquence existante. Dans le cas de longues séries de prédiction, de l’incertitude apparaît et s’accroît à mesure que le nombre de prédictions augmente. Certains travaux cherchent à réduire cette incertitude en se spécialisant sur cette problématique [52]. Cependant, les méthodes de reconstruction peuvent bénéficier du fait qu’il existe des données réelles après une prédiction. Cette particularité doit être exploitée pour réduire le nombre de valeurs probables, en utilisant les données connues comme des repères dans les prévisions.

I.2.2.2 A base de réseaux de neurones récurrents

Au milieu de ce vaste état de l’art, les RNRs demeurent une certitude et une excellente base de travail pour la prédiction [53], la reconstruction et toutes les tâches qui impliquent une dimension temporelle, ou de manière générale séquentielle. Bon nombre d’applications concluantes de leur utilisation pour la prédiction et la prédiction de grandeurs physiques constituent une large littérature, surtout dans le domaine temporel [54]–[60], parfois dans le spatio-temporel [61], [62].

Avant que les modèles d’AP émergent, des modèles d’analyse statistiques comme les filtres de Kalman ou les modèles autorégressifs et moyenne mobile étaient utilisés pour prédire les états suivants des séries temporelles et spatio-temporelles. Le modèle AutoRegressive Integrated Moving Average (ARIMA) [63] est devenu une référence comparative pour les méthodes de prévision d’AP [53], [64]. En 2023, les performances de ces modèles sont dépassées par les applications d’AP aux STs [65]. Les modèles de type Fractionnaly AutoRegressive Integrated Moving Averages (FARIMAs) [66] offrent une généralisation des modèles ARIMAs pour tenir compte des processus à mémoire longue en introduisant un ordre de différenciation fractionnaire. Bien que les prédicteurs basés sur ce type de modèle obtiennent de bons résultats pour certaines applications et avec certaines conditions d’usage [67], [68], les méthodes d’AP restent, en général, des solutions plus versatiles et performantes. Elles sont en particulier plus performantes à des échelles de temps fines [67], lorsqu’il faut prédire des valeurs en dehors des instants de la grille des échantillons de temps [68], pour des signaux à forte dynamique [69] et également pour des signaux de température [70]. De nombreuses solutions d’hybridation FARIMA-RNR ont par ailleurs été proposées, en particulier dans le domaine financier [71], afin de bénéficier du meilleur des deux approches. Néanmoins, les modèles autorégressifs présentent pour nous un inconvénient majeur qui est que par construction ils requièrent que les données soient échantillonnées à des pas de temps fixe, ce qui n’est pas compatible avec nos hypothèses de travail. C’est pourquoi notre étude se focalise uniquement sur les méthodes d’AP.

Yunbo Wang et son équipe ont développé une cellule neuronale récurrente innovante [72] qui traite également des données spatiales, ils l’ont appelé Spatio-Temporal Long Short-Term Memory (ST-LSTM). Elle est basée sur la structure d’une cellule LSTM. Ils ont amélioré et éprouvé cette technologie au fil des années pour finalement surpasser les méthodes de l’état de l’art dans la prédiction de mouvements avec leur modèle PredRNN [73], [74] (notamment de trajectoires GPS, d’images radar, de vidéos d’actions). La cellule ST-LSTM propage plus efficacement les dynamiques spatiales dans l’état de la cellule que la ConvLSTM [40], sa prédécesseuse.

I.2.2.3 Diverses solutions

Dans le cas de prévision de longues séquences, l’incertitude s’accroît avec les prédictions (de rares applications ont besoin de prévoir loin dans le futur). Le modèle Informer [52] est un modèle qui se spécialise dans ce genre de prévision longue distance. Il est adapté du Transformeur originel, sans trop de modifications, pour la

réalisation de ses objectifs. Ce travail met en lumière la capacité du Transformeur à être plus efficace qu’un réseau LSTM pour les prédictions éloignées.

Le cas de données raster amène une problématique supplémentaire, celle de l’exploitation des corrélations spatiales. A ce jour, les RNCs demeurent les mieux conçus pour apprendre ces corrélations [75]. Ce cas est compliqué car il nécessite un RN qui combine une architecture récurrente ou capable d’un exercice de mémoire et une architecture convolutive au mieux. Par exemple, ces travaux ont une approche purement convolutive [76]–[78] tandis que celui-ci a une approche purement récurrente [61] et celui-là vectorise la sortie d’un RNC pour l’entrée d’un RNR [62]. Tous les auteurs de ces travaux expliquent dans leur papier que la combinaison des RNRs et RNCs est un challenge de l’AP rendu difficile par les représentations de données fondamentalement différentes des premiers et des seconds. D’autres travaux contournent le problème en manipulant les données différemment pour s’adapter à d’autres RNs. Par exemple : celui-ci [79] combine une analyse en composantes principales pour modifier la base de représentation des données raster et un RNPA pour les traiter; celui-là [80] représente les données dans un graphe pour modéliser les corrélations spatiales et décompose les signaux temporels avec une transformée en ondelettes, puis un Transformeur effectue les prédictions qui sont incorporées comme caractéristiques dans les noeuds du graphe pour être propagées et ré-estimées par un RNG; cette approche [81] effectue deux prévisions, une dite locale à partir d’une ST et une dite globale à partir de l’ensemble des STs pour les factoriser avec un autre modèle basé sur de la convolution.

Cette diversité au sein de la littérature signifie qu’aucun modèle d’AP n’est généralisable à l’ensemble des applications de prévision de données raster. Les méthodes qui sont adoptées sont propres aux problématiques et aux caractéristiques des données.

I.2.3 Imputation de données manquantes

I.2.3.1 Introduction

La présence de données manquantes dans un jeu de données est monnaie courante dans presque tous les domaines. Cela peut être dû à des dysfonctionnements de l’équipement qui génère les données. Mais le plus souvent cela est dû à la compilation de données issues de sources différentes, de dates différentes, de systèmes différents ou à l’ajout de caractéristiques non partagées par l’ensemble des données. Quelle que soit la source du problème, il faut impérativement combler ce manque pour traiter la donnée avec des RNs. Les points d’entrées des RNs n’ont pas de dimensions dynamiques, un jeu de données doit respecter des dimensions fixes.

L’imputation de données manquantes est donc un besoin même au sein des applications d’AP. Déterminer les données manquantes est un pari risqué lorsqu’on utilise des RNs car on peut biaiser l’apprentissage en induisant le réseau en erreur si les valeurs imputées sont hasardeuses. La méthode la plus intuitive, néanmoins efficace, est d’uniformiser toutes les valeurs manquantes à une valeur prédéfinie qui aura, pour le RN, la signification de valeur manquante. Fréquemment, cette valeur sera -1 dans un jeu de données normalisées (mise à l’échelle bornée entre 0 et 1) ou 0 dans un jeu de données standardisées (moyenne nulle et écart-type de 1). Autrement, si les valeurs imputées sont suffisamment probables, cela peut être un atout non négligeable pour améliorer les performances du RN [82]. Les méthodes d’imputation de données manquantes à partir d’AP sont réputées pour accomplir ce type de tâche.

La tâche de reconstruction est plus proche de celle de l’imputation de données manquantes que celle de la prévision de données dans le sens où une ST, ou spatio-temporelle, compressée peut prendre la forme d’un jeu de données avec des données manquantes. La résolution de ce problème de reconstruction est favorisée par le fait que le jeu de données d’entraînement est, en théorie, exempt de données manquantes; contrairement à une application standard d’imputation.

I.2.3.2 Aux séries temporelles multivariées

Les données manquantes au sein de ces données sont un problème très fréquent et la recherche de la meilleure solution pour leur imputation est la tâche connexe à ce travail de thèse la plus explorée de ces dernières années. Une ST multivariée est une série temporelle qui comporte plusieurs caractéristiques, c’est majoritairement le cas

des signaux des RCSFs. Par exemple, un signal provenant d’un nœud de capteur peut comporter des informations différentes comme la température, l’humidité, la luminosité, etc; constituant une ST multivariée.

Dans ce domaine, les méthodes à base de RAG sont très populaires [83]–[87] pour leur aspect génératif qui permet de créer de toute pièce de la donnée réaliste en s’appuyant sur des informations réelles. Étant donné le cadre de l’imputation qui suppose que le jeu de données d’entraînement contienne des données manquantes, la non-nécessité des méthodes antagonistes génératives à devoir comparer un jeu de données complet à un jeu de données incomplet pour valider les données générées en fait un atout indéniable. Cependant, ils sont sous-représentés dans le domaine spatio-temporel [88], spécialement pour l’imputation de données manquantes. Ils requièrent de plus amples améliorations pour stabiliser l’entraînement, en particulier pour modéliser les données spatiales et spatio-temporelles et capter efficacement les corrélations spatiales.

Encore une fois, il n’existe aucun consensus autour d’un modèle fédérateur puisque les modèles de diffusion, les Transformeurs et les AEs sont également employés à imputer des données manquantes aux séries temporelles multivariées [89]. Il apparaît récemment que le modèle de diffusion soit plus efficace que les AEs [90], [91]. La raison principale est qu’un AE récurrent nécessite bien souvent une couche de répétition de vecteur [92] pour reformer la taille de la séquence initiale considérablement réduite, ce qui cause une perte d’information et un biais d’apprentissage. L’astuce avec les modèles de diffusion est d’assimiler les données manquantes à du bruit. Tandis que ce modèle plus récent [93] dispose d’un encodeur proche du Transformeur avec de l’auto-attention éparse pour éviter de considérer les données manquantes comme informations pour l’espace latent. Il obtient également des performances concurrentielles. Les Tableaux I.1, I.2 et I.3 recourent les résultats expérimentaux de quelques uns de ces différents travaux sur le jeu de données Physionet (<https://physionet.org/content/challenge-2012/1.0.0/>). C’est un jeu de données médical publique de STs essentiellement constitué de bilans sanguin de patients mourants admis à l’hôpital. Il présente 80,67% de données manquantes qu’il faut imputer. Les tableaux I.1, I.2 présentent des résultats de perte de régression sur les caractéristiques du jeu de données Physionet. Le tableaux I.3 présente les scores de réussite sur la prédiction de mortalité des patients à partir du jeu de donnée complété par les modèles de RN. Le modèle de RNR bidirectionnel BRITS [94] est une base de comparaison récurrente pour l’imputation de données manquantes à des STs.

Modèle	Référence	Taux de données manquantes				
		10%	30%	50%	70%	90%
BRITS	[94]	0.611	0.672	0.779	0.809	0.850
E ² GAN	[83]	0.679	0.739	0.789	0.823	0.853
E ² GAN-RF	[84]		0.659	0.714	0.756	0.831
SSGAN	[85]	0.598	0.670	0.762	0.782	0.818
ST-Impute	[93]	0.547	0.603	0.702	0.720	0.764

Table I.1: Comparaison de performances pour la régression sur le jeu de données Physionet en fonction du taux de données manquantes (en racine d’erreur quadratique moyenne)

Modèle	Référence	Taux de données manquantes				
		10%	30%	50%	70%	90%
BRITS	[94]	0.278	0.303	0.336	0.361	0.384
MARNN	[86]	0.261	0.297	0.331	0.358	0.380
CSDI	[90]	0.217		0.301		0.481

Table I.2: Comparaison de performances pour la régression sur le jeu de données Physionet en fonction du taux de données manquantes (en erreur moyenne absolue)

Modèle	Référence	Score
BRITS	[94]	0.837
E ² GAN	[83]	0.872
E ² GAN-RF	[84]	0.883
ST-Impute	[93]	0.850
MARNN	[86]	0.863
SSGAN	[85]	0.851

Table I.3: Comparaison de performances de prédiction de mortalité sur le jeu de donnée Physionet (en score AUC)

Ces comparaisons sont données à titre indicatives mais ne peuvent pas supposées à elles seules d’un classement en toutes circonstances et pour des jeux de données

différents. Ces travaux révèlent néanmoins certaines affirmations :

- Bien que ne présentant aucun résultat sur le jeu de données Physionet, le modèle SSSD [91] serait, d’après ces auteurs et leurs expérimentations, plus performant que le modèle CSDI [90]. Faisant de lui le meilleur modèle de diffusion à l’état de l’art 2023 pour cette tâche.
- Le modèle BRITS [94] est unanimement dépassé, la comparaison avec ce modèle n’est plus pertinente.
- Le modèle MARNN [86] est le plus récent et le plus performant des RAGs de cette comparaison.
- Le modèle ST-Impute [93] est le plus récent et le plus performant des modèles avec mécanisme d’attention de cette comparaison.
- Le modèle E²GAN-RF [84] attire l’attention pour une tâche de reconstruction. Son hybridation avec un AE et la partie « real-data forcing » de son architecture lui donne l’habileté intrinsèque à tendre moins vers un générateur créatif réaliste et plus vers un générateur reconstituteur. Sa performance sur le score de prédiction de mortalité, Tableau I.3, qui est une information complète du jeu de données va dans ce sens.

I.2.3.3 Aux données raster

Pour rappel : un ensemble de signaux d’un RCSF est catégorisé comme données raster selon cette étude [3] car les signaux temporels sont collectés à partir de positions fixes et irrégulièrement distribuées dans un espace défini. Il n’existe pas de consensus à propos d’un choix de modèle adapté à ce genre de données dans la littérature. Il y a bon nombre de choix de représentations de données d’entrée et d’architectures de RNs différentes. En d’autres termes, les données raster peuvent être instanciées en tant que séquences, graphiques, matrices ou tenseurs, et apprises par des RNPAs, RNRs, RNCs, RNGs et autres combinaisons de ces derniers tels que les ConvLSTM ou les GraphCNN.

L’état de l’art en matière d’imputation de données raster manquantes propose toutes sortes de modèles, mais pour ce cas les RAGs se montrent moins performants [88]. En 2023, les modèles de diffusion se démarquent dans cette tâche, en particulier le modèle PriSTI [95] qui utilise le mécanisme d’auto-attention deux fois dans son décodeur pour observer l’élément courant dans sa séquence temporelle et dans sa séquence

spatiale. Il combine également ces corrélations croisées avec des caractéristiques géographiques directement contenues dans la donnée. Ce modèle tire profit des travaux sur le modèle CDSI [90] et le modèle GRIN [96].

Le modèle PredRNN [74], qui capture mieux les corrélations spatiales et temporelles que le modèle ConvLSTM [40], présente une cellule spatio-temporelle ST-LSTM qui doit être particulièrement intéressante à exploiter pour réaliser de l’imputation de données manquantes. Elle pourrait être utilisée dans les couches d’un modèle d’AE ST-LSTM dédié à l’imputation.

Les travaux sur le modèle DeepMDP de Kök et Özdemir [97] proposent un RNR pour imputer des données manquantes au sein d’un RCSF. Grâce à son architecture simple d’une à trois couches LSTMs de quelques centaines de cellules cachées, le modèle DeepMDP effectue des calculs rapides et demande suffisamment peu de ressources pour pouvoir être embarqué sur les nœuds d’un RCSF. Leur protocole est utilisé pour compléter les données transmises au niveau des nœuds tandis que nos travaux permettent de compresser l’acquisition des données à ce même niveau puis de les compléter sur le serveur central, ce qui implique un traitement global des signaux et différents algorithmes. Bien que le modèle DeepMDP soit récent dans le domaine des RCSFs, son application et ses objectifs sont trop éloignés de ceux poursuivis par cette thèse.



Développement de méthodes de reconstruction des signaux comprimés par apprentissage profond – application au réseau de capteurs

Sommaire

II.1	Méthodologie	60
II.1.1	Méthodes de reconstruction	60
II.1.1.1	Introduction	60
II.1.1.2	Représentation des données	62
II.1.1.3	Méthode temporelle	65
II.1.1.4	Méthode spatiale	67
II.1.1.5	Méthode spatio-temporelle	70
II.1.2	Les algorithmes de traitement du signal	73
II.1.2.1	Estimation de la fiabilité statistique d'une prédiction	73

Chapitre II – Développement de méthodes de reconstruction des signaux comprimés par apprentissage profond – application au réseau de capteurs

II.1.2.2	Régularisation d'un signal prédit	74
II.1.2.3	Discrimination des erreurs de prédiction	75
II.1.3	Création d'une API dédiée	77
II.1.3.1	Présentation de deepsignals	77
II.1.3.2	La classe Application	77
II.1.3.3	Les classes de profils	78
II.1.3.4	La classe Dataset	79
II.1.3.5	Les autres classes	79
II.2	Applications	81
II.2.1	Les ressources d'expérimentation	81
II.2.1.1	Logicielles et matérielles	81
II.2.1.2	Le réseau de capteur sans fil	81
II.2.1.3	Le jeu de données	83
II.2.2	Application de la méthode temporelle	87
II.2.2.1	Le modèle et l'entraînement des prédicteurs	87
II.2.2.2	Les résultats	88
II.2.3	Application de la méthode tempo-spatiale	94
II.2.3.1	Les modèles	94
II.2.3.2	L'entraînement	98
II.2.3.3	Les résultats	100
II.3	Bilan	110

Les travaux de thèse débutent en ayant pour objectif d'améliorer les performances parues dans la thèse de Frank Itoua-Engoti [9] d'acquisition et de reconstruction de signaux de la méthode ACK [10] qui est appliquée aux RCSFs. La conception d'une nouvelle méthodologie impliquant l'AP est au coeur de ce chapitre et son application aux signaux compressés d'un RCSF concrétise la contribution de ces travaux à la littérature scientifique.

II.1 Méthodologie

Cette méthodologie traite des signaux complets continus selon un pas de temps constant librement paramétrable selon le niveau de finesse que l'on veut observer. Cela implique que chaque valeur réelle acquise doit être synchronisée sur un horodatage que le pas de temps connaît.

II.1.1 Méthodes de reconstruction

II.1.1.1 Introduction

La théorie de l'AC suppose qu'un signal peut être représenté avec parcimonie dans différentes bases de représentation. Ce point est important parce qu'il implique qu'il existe, pour chaque type de signal, une base de représentation optimale sur critère de parcimonie. De plus, les matrices de détection utilisées en AC sont des matrices aléatoirement établies par une distribution uniforme car elles permettent d'obtenir des taux de compression plus élevés que la plupart des matrices de détection déterministes connues. Or, une construction aléatoire de cette matrice entraîne un risque de non-répétabilité de la qualité de reconstruction d'un même signal. Cela suppose aussi qu'il existe une matrice de détection optimisée pour échantillonner un signal qui permette d'atteindre le nombre de mesures minimales pour une précision de reconstruction donnée au sein d'une base de représentation. La notion de précision désigne l'exactitude d'une reconstruction par l'évaluation de l'erreur qui existe entre les valeurs d'un signal d'origine complet et son estimation à partir d'une version compressée de lui-même. On appelle un signal compressé, un signal incomplet qui est discrétisé de façon irrégulière à un faible taux et pour lequel il faut imputer les échantillons manquants à un taux au moins égal à celui de Nyquist. Si une méthode d'AP peut trouver la matrice de détection et la base de représentation optimales pour l'acquisition et la reconstruction d'un signal de RCSF par AC alors l'augmentation de son taux de compression, c'est-à-dire

le rapport $\rho = \frac{n}{m}$ (pour rappel: n la dimension du signal, m le nombre de mesures échantillonnées), peut s'opérer tout en maintenant une reconstruction précise du signal.

L'idée précurseuse à l'origine des travaux de thèse qui en découle permettrait de transformer une matrice de détection aléatoire en matrice de détection déterministe pour un signal donné en suivant les étapes illustrées Figure II.1. Ces opérations consistent en une simulation d'acquisition et de reconstruction d'un signal d'un RCSF assistée par un RN en vue d'augmenter son taux de compression.

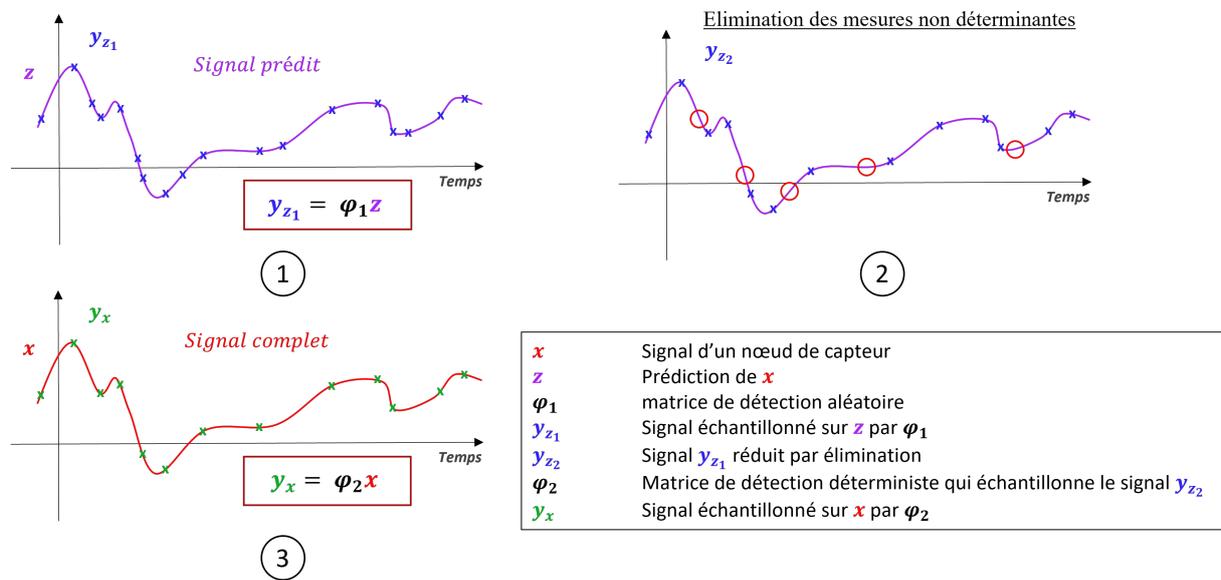


Figure II.1: Théorie de l'optimisation d'une matrice de détection aléatoire d'AC à l'aide d'AP

Supposons connaître une matrice de détection aléatoire φ_1 destinée à être utilisée pour acquérir le nombre de mesures m d'un prochain signal y_x représentant une version compressée d'un signal x de longueur n sur le RCSF. Considérons le signal z qui est une prévision du signal x généré par un RN. Le signal compressé y_z résulte de φ_1 et de z selon la formule 1.2 de l'AC:

$$y_{z_i} = \sum_{j=1}^n \varphi_{1,i,j} z_j, \quad i = 1, \dots, m$$

On souhaite alors mettre au point un traitement qui calculerait plusieurs reconstructions de z par le biais des algorithmes de reconstruction de l'AC tout en réduisant progressivement le nombre de mesures du signal compressé y_z de telle sorte à ne conserver que les mesures déterminantes et significatives pour la reconstruction du signal. Ainsi,

il existe φ_2 , la matrice de détection issue de φ_1 produisant un nombre de mesure l utile à la reconstruction de z tel que $l < m$. Cette méthode permettrait d'éliminer les acquisitions aléatoires non déterminantes pour la reconstruction. Alors, considérant que x et z sont de même nature, la matrice de détection φ_2 supplanterait φ_1 pour commander l'échantillonnage du signal compressé y_x au RCSF tel que :

$$y_{x_i} = \sum_{j=1}^n \varphi_{2_{i,j}} x_j, \quad i = 1, \dots, l$$

L'idée suppose au préalable qu'un modèle de RN est capable d'établir la prévision z d'un signal x à échantillonner avec précision. En investiguant la littérature, nous avons commencé par entraîner un RNR à la résolution de ce problème. Les prévisions de signaux Z qu'il accomplissait étaient à ce point précis en comparaison avec les signaux X du RCSF que nous avons expérimenté une reconstruction de signaux comprimés par imputation de prédictions du RNR aux mesures manquantes. Il est apparu que les résultats de cette méthode de reconstruction étaient plus performants sur critère de précision et de vitesse de calcul que la méthode ACK appliquée aux mêmes signaux comprimés pour des taux de compression des signaux inférieurs à 85%. Il devenait alors intéressant d'approfondir ce travail pour vérifier si une méthode de reconstruction d'AP basée sur la prévision était capable de surpasser les algorithmes de reconstruction de l'AC. Dans un premier temps, cette piste a conduit au développement d'une méthode de reconstruction temporelle de signaux comprimés qui surpasse en précision celle de l'ACK et celles de l'état de l'art de l'AP. Dans un second temps, l'exploitation des corrélations spatiales qui existent entre les nœuds de capteur d'un RCSF et leurs signaux a permis de concevoir une méthode de reconstruction spatio-temporelle qui surpasse la méthode temporelle en matière de précision. L'idée précurseuse, qui est présentée ici, à l'origine de ces méthodes est devenue une perspective de ces travaux.

II.1.1.2 Représentation des données

L'état de l'art décrit au chapitre I montre que le choix de représentation des données conditionne le choix du RN à modéliser, la réciproque est aussi vraie. Les travaux menés mettent en collaboration les RNRs avec les RNCs pour reconstruire un ensemble de signaux. Cela implique que les représentations des données sont adaptées à ces classes de RNs. Dans le cas d'une reconstruction temporelle, effectuée par RNR, les signaux sont instanciés sous forme d'un tenseur 3D au sens de l'AP (cf I.2.3.3). Cette

représentation permet de décrire des STs et elle est tout à fait commune et adaptée aux RNRs. Les axes de ce tenseur sont : le nombre de signaux, le temps et le nombre de caractéristiques, comme illustré sur la Figure II.2.

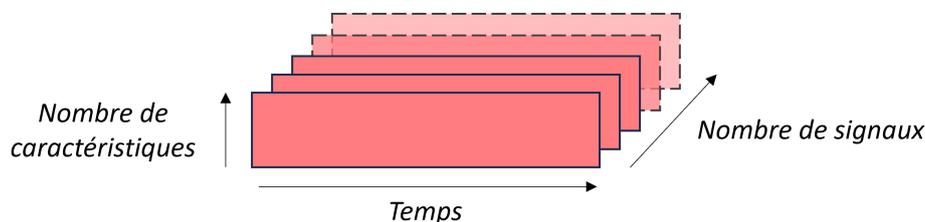


Figure II.2: Illustration d'un tenseur 3D pour les signaux d'un RCSF

Dans le cas d'une reconstruction spatiale, effectuée par RNC, les signaux sont instanciés sous forme d'un tenseur 5D. Les axes de ce tenseur sont : le temps, la largeur, la hauteur, la profondeur et le nombre de caractéristiques, comme illustré sur la Figure II.3.

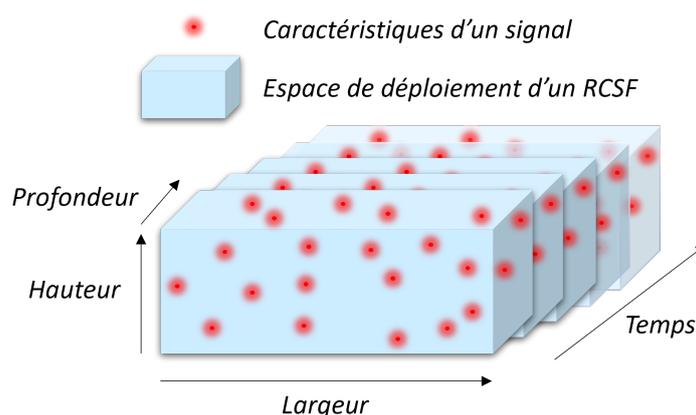


Figure II.3: Illustration d'un tenseur 5D pour les signaux d'un RCSF

Ce tenseur permet de représenter les positions des nœuds d'un RCSF dans un espace en 3 dimensions. Dans des termes plus concrets, tous les instants des signaux temporels acquis par les capteurs sont compilés en une liste de cartes en 3 dimensions de ces valeurs. Une carte est un tableau en 3 dimensions dont les tailles correspondent avec le lieu d'installation du RCSF (comme un bâtiment, un parc, une ville, etc...) selon une unité de distance prédéfinie. On pourrait aussi représenter un RCSF par un tenseur de plus faible dimension comprenant seulement les axes suivants: le temps, le nombre de signaux et le nombre de caractéristiques. Or, des informations telles que les distances entre les nœuds de capteurs, leur élévation, ou l'architecture de l'espace d'installation, qui sont des informations importantes contenues dans une représentation en 3 dimensions de l'espace pour décrire des corrélations spatiales entre des signaux,

manquent dans ce dernier tenseur. On exclut donc cette forme de représentation de nos méthodes dédiées aux données raster.

Dans la littérature, les données spatiales et spatio-temporelles le plus fréquemment utilisées avec des RNs, spécialement les RNCs, sont des images et des vidéos. La différence majeure entre ces données et un RCSF est que les données spatiales d'un RCSF présentent un grand nombre d'espaces « vides » tenant au fait que les nœuds de capteurs sont éparpillés irrégulièrement dans l'espace. C'est un aspect important de la donnée qui doit être appréhendable par le RNC pour préserver la structure du RCSF durant la phase d'entraînement et ainsi conduire à une reconstruction des signaux qui sera contrainte par cette structure. Les espaces vides ne doivent pas être interprétés par le RN comme des données réelles ou significatives et doivent également se distinguer des données manquantes.

Une donnée manquante au sein d'un signal compressé est tout simplement ignorée par la méthode de reconstruction temporelle qui ne traite que des valeurs réelles ou générées (la méthode est expliquée dans la sous-section II.1.1.3 suivante). Or, cette facilité ne peut s'appliquer à une représentation spatiale dans laquelle peuvent se trouver, à la fois, des valeurs réelles et manquantes. Pour illustrer cet état de fait, prenons l'exemple d'un ensemble de signaux compressé à 50%, alors 50% des valeurs sont manquantes et 50% des valeurs sont réelles. Si on considère seulement les valeurs d'un point de l'échelle du temps, il y aura à la fois des valeurs réelles acquises par certains nœuds de capteur et des valeurs manquantes non acquises par des nœuds de capteurs en veille à ce moment-là. La représentation en 3 dimensions de ces valeurs comprendra alors des espaces sans capteurs et des espaces sans mesures de capteurs. Le choix d'imputation de valeurs à ces espaces vides pour le RNC doit être réalisé conformément à leur cause. Même en considérant des nœuds de capteurs mobiles, un espace vide demeure un espace vide. Il doit être représenté par une valeur constante. En se basant sur l'état de l'art, cette valeur est -1 dans le cas d'une normalisation des données (mise à l'échelle bornées entre 0 et 1), de façon à exclure un espace vide des valeurs significatives, ou 0 dans le cas d'une standardisation des données (distribution uniforme de moyenne nulle avec un écart-type de 1). La représentation d'une donnée manquante est effectuée de façon similaire, de telle sorte que le RN assimile qu'une valeur unique a pour signification que la valeur est manquante. Afin de réaliser une distinction et d'informer le RN des positions des nœuds de capteur, on ajoute une caractéristique binaire aux données, appelée simplement « capteur », qui indique 0 si c'est un espace vide ou 1 si c'est une mesure

de capteur manquante ou une valeur réelle.

Dans le cas d'une reconstruction spatio-temporelle cette représentation des données porte des informations supplémentaires qui améliorent l'apprentissage des RNCs. Une valeur manquante est générée par un RNR et la caractéristique capteur indique une probabilité de véracité de la valeur générée à la place de 1 si c'est une mesure de capteur manquante. Cet aspect de la représentation spatio-temporelle est développé dans la section II.1.1.5 dédiée à cette méthode.

II.1.1.3 Méthode temporelle

La reconstruction repose sur l'emploi de RNRs. L'état de l'art a prouvé que ce moyen garantit de bons résultats et se trouve être parfaitement adapté à des STs tels que les signaux d'un RCSF. Il convient de rappeler que cette classe de RN est naturellement conçue pour prédire les valeurs suivantes à partir d'une ST passée en entrée du RNR. Néanmoins, elle est incapable d'imputer des valeurs manquantes dans une séquence à moins d'être modélisée en AE. Or, l'état de l'art en 2023 montre que ce modèle n'est pas le plus efficace dans l'accomplissement de cette tâche. C'est pourquoi on préfère établir une reconstruction temporelle par une approche prévisionnelle (cf I.2.2) des données manquantes et non par imputation dans le sens de l'état de l'art (cf I.2.3).

Contrairement aux travaux de la littérature qui effectuent généralement des traitements séquence-à-séquence [51], ou Sequence-to-sequence (Seq2seq), notre méthode effectue un traitement point-à-point. La raison principale qui motive ce mode de reconstruction est due au sous-échantillonnage aléatoire des signaux compressés, ils ne sont naturellement pas constitués de séquences de valeurs indexées par un pas de temps régulier et suffisamment fin. En d'autres termes, les valeurs des signaux compressés sont temporellement isolées les unes des autres. Les séquences d'entrée des modèles de RNRs de notre méthode sont donc de longueur 1. Ce mode d'acquisition comprimée correspond également avec une technologie émergente dans l'IoT appelée acquisition intermittente [98]. Elle se base sur la partie matérielle des objets connectés qui s'alimentent en récupérant l'énergie environnante telle que l'énergie solaire ou mécanique du vent et de l'eau. Ainsi, des noeuds de capteurs effectuent l'acquisition et l'envoi d'une mesure lorsque l'énergie nécessaire est récupérée. Cette avancée technologique pour l'écologie et l'économie reste dépendante de l'environnement et engendre alors des signaux comprimés à des pas de temps irréguliers. Notre méthode de traitement point-à-point est parfaitement adaptée pour traiter des données

acquises par intermittence.

En partant d'une valeur réelle d'un signal compressé, le modèle entraîné calcule alors la prédiction de la valeur suivante puis recommence l'opération en partant de cette prédiction jusqu'à la prochaine valeur réelle du signal compressé. Un signal reconstruit par ce procédé est appelé un signal prédit. Une prédiction ne peut pas garantir une précision de 100%, c'est le principe même d'une prédiction, c'est une estimation. Alors, l'inconvénient d'un tel procédé est que l'incertitude s'accumule à chaque nouvelle prédiction. Les valeurs prédites deviennent de moins en moins fiables à mesure que la distance temporelle qui les séparent d'une valeur réelle augmente. Il est alors intéressant d'élaborer une reconstruction collaborative à partir de deux signaux prédits causalement et anti-causalement. La figure II.4 illustre le phénomène croissant d'incertitude à travers le temps en montrant que le croisement de deux signaux prédits inversement l'un de l'autre dans l'intervalle de deux valeurs réelles réduit les probabilités et donc des erreurs de prédictions.

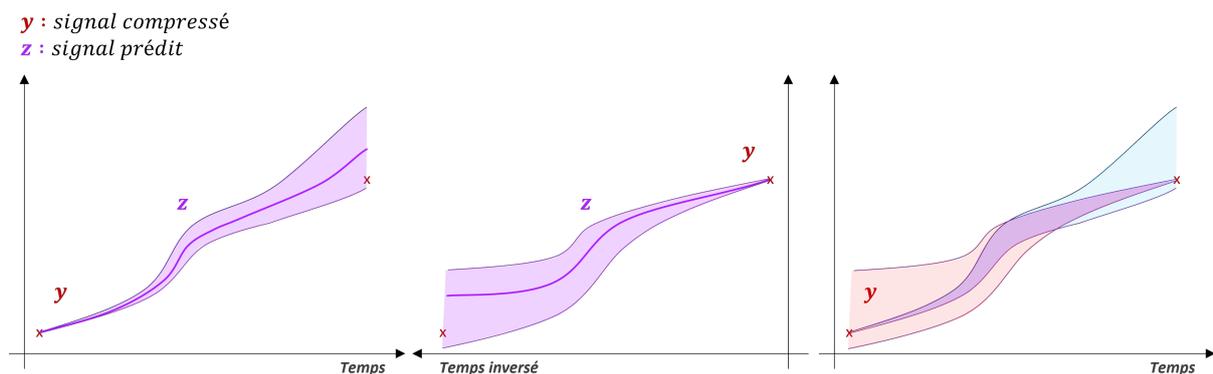


Figure II.4: Préviation d'un signal: marge d'erreur croissante dans le temps

Notre approche est donc basée sur l'usage d'un couple de modèles récurrents entraînés par le même jeu de données et qui fonctionnent à l'inverse temporel l'un de l'autre, voir Figure II.5. C'est-à-dire que l'un des RNRs produit une prédiction causale des échantillons à reconstruire à partir des échantillons antérieurs réellement mesurés. Tandis que le second RNR fait progresser ses prédictions anti-causalement. Ce mode de reconstruction ne peut s'effectuer uniquement dans un intervalle de deux valeurs réelles. Pour un usage en temps réel, le signal prédit causalement prédomine au delà de la dernière mesure échantillonnée.

L'association de ces signaux se fait par une moyenne pondérée par la fiabilité statistique w de chaque prédiction, c'est-à-dire, la probabilité que cette prédiction corresponde à la

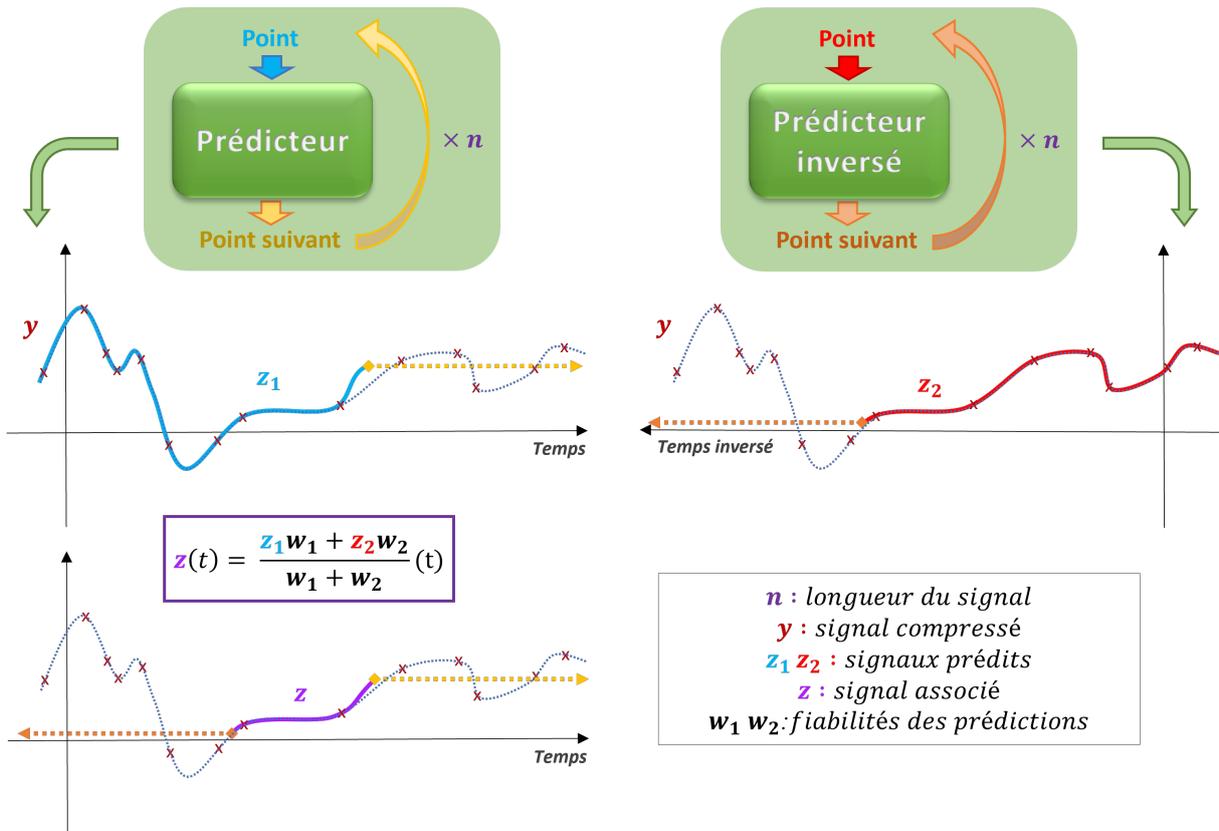


Figure II.5: Reconstruction temporelle collaborative causale et anti-causale

valeur réelle. On calcule cette probabilité en considérant une marge d'erreur acceptable et paramétrable en fonction du niveau d'exactitude désiré pour une reconstruction. Différents tests ont confirmé que cette méthode conduit à une reconstruction du signal compressé plus précise qu'une simple moyenne des signaux prédits. Concrètement, considérant une valeur réelle du signal, la valeur reconstruite qui la précède sera plus proche du signal prédit anti-causalement tandis que la valeur reconstruite qui la suit sera plus proche du signal prédit causalement.

II.1.1.4 Méthode spatiale

La reconstruction spatiale, illustrée Figure II.6, est assurée par des RNCs modélisés en AE. Cette modélisation des RNCs est fédératrice au sein de la littérature (cf I.2.3.3, I.2.2.3). Pour chaque point du domaine temporel, une carte en 3 dimensions de données raster, qui inclut des données manquantes dans ce cas, est apprise et encodée par un premier modèle de RNC dans une représentation interprétable par un second modèle de RNC qui, en décodant cette carte, la complète.



Figure II.6: Reconstruction spatiale réalisée par auto-encodage

La difficulté majeure et spécifique à cette tâche est d'encoder des données qui peuvent être irrégulièrement compressées, c'est-à-dire, que les taux de compression varient d'une carte à une autre. C'est la raison pour laquelle il n'est pas possible de représenter un ensemble de signaux comprimés uniquement par ses valeurs réelles. En effet, mis à part le nombre d'échantillons, un RNC nécessite un tenseur d'entrée aux dimensions statiques, tant durant son apprentissage que son utilisation.

Pour relever ce défi, plusieurs modèles d'AEs ont été développés et testés. Il est apparu au travers des résultats qu'un modèle pouvait être plus performant dans l'ensemble mais pas systématiquement le meilleur reconstruteur pour chaque échantillon de carte de données. Nous avons donc choisi de conserver trois modèles d'AEs convolutifs pour réaliser la reconstruction spatiale par apprentissage ensembliste. Chaque architecture d'AE met en œuvre une stratégie de reconstruction différente pour identifier les corrélations spatiales existantes entre les signaux et en déduire les valeurs des mesures manquantes.

Le modèle d'AE à réduction de dimensionalité réduit la représentation de la donnée d'entrée en compressant ses dimensions spatiales lors de l'encodage. Les dimensions d'entrée de l'encodeur et de sortie du décodeur correspondent tandis que les dimensions de sortie de l'encodeur et d'entrée du décodeur correspondent et sont plus petites. Cette technique est largement employée dans des travaux de compression et de décompression de données destinées à la transmission ou au stockage pour alléger les coûts énergétiques, de communication et de consommation de mémoire.

Le modèle d'AE à large spectre considère un grand nombre de données voisines pour établir la reconstruction d'un point dans l'espace. De cette façon la compréhension des corrélations spatiales est renforcée dans sa globalité. Le moyen d'y parvenir est d'augmenter considérablement la taille du champ récepteur (cf I.1.4.4). Ceci a aussi pour effet d'augmenter la taille du noyau de convolution ce qui augmente aussi la

complexité du RNC. Sans pour autant coûter autant de ressources qu'un Transformeur, ce modèle est le plus gourmand des trois en temps d'exécution, d'entraînement et en consommation de mémoire.

Le modèle d'AE variationnel traduit la donnée incomplète d'entrée en paramètres d'une distribution statistique Gaussienne : une moyenne et une variance. Cette représentation suppose qu'il existe un espace latent continu qui structure les variations spatiales des signaux d'un même ensemble. Ce modèle apprend cette structure d'espace latent. La stochasticité d'une telle distribution améliore en réalité sa robustesse et exige de l'AE qu'il apprenne des représentations significatives valides pour chaque point de l'espace latent.

Durant le développement de la méthode spatiale, nous avons aussi mis au point plusieurs modèles de RAGs pour tenter de surpasser le niveau de précision de reconstruction des AEs. Cependant les nombreux essais réalisés ont confirmé ce que la littérature avait déjà mis en lumière [88]: l'apprentissage d'un RAG à partir d'une information de sortie binaire entre donnée réelle et falsifiée est instable et limite sa compréhension d'un jeu de données d'entraînement exempt de données manquantes. De plus, l'étude de l'état de l'art a montré que ce modèle n'est pas encore compétitif dans l'exploitation des corrélations spatiales. Les erreurs de reconstruction issues de ces expérimentations se sont toujours situées au-delà de celles des AEs.

La méthode spatiale adoptée établit une reconstruction de carte à partir de trois autres, chacune réalisée par un AE différent. Chaque stratégie de reconstruction mise en oeuvre présente des avantages mais aussi des limites en fonction des situations de compression de données, elles sont illustrées sur la Figure II.7.

Nous avons observé qu'il n'existe pas d'AE systématiquement meilleur que les autres pour reconstruire tous les échantillons de cartes. Afin de tirer le meilleur de ce tripartite pour une reconstruction finale plus précise, nous avons commencé par effectuer une moyenne simple de leurs résultats et, déjà, une reconstruction spatiale moyennée s'est montrée plus précise que les reconstructions individuelles des AEs sur l'ensemble des signaux de RCSF de tests. Les expérimentations de reconstruction spatiale réalisées par un RAG ont inspiré une façon de pondérer la contribution de chaque AE: on utilise un réseau discriminatoire pour établir le pourcentage de vraisemblance de chaque carte reconstruite par un AE. Ce discriminateur reçoit en entrée de son réseau des cartes de données et les classe dans les catégories carte reconstruite par un AE ou carte complète de valeurs réelles. En l'entraînant à réaliser cette discrimination, il estime en

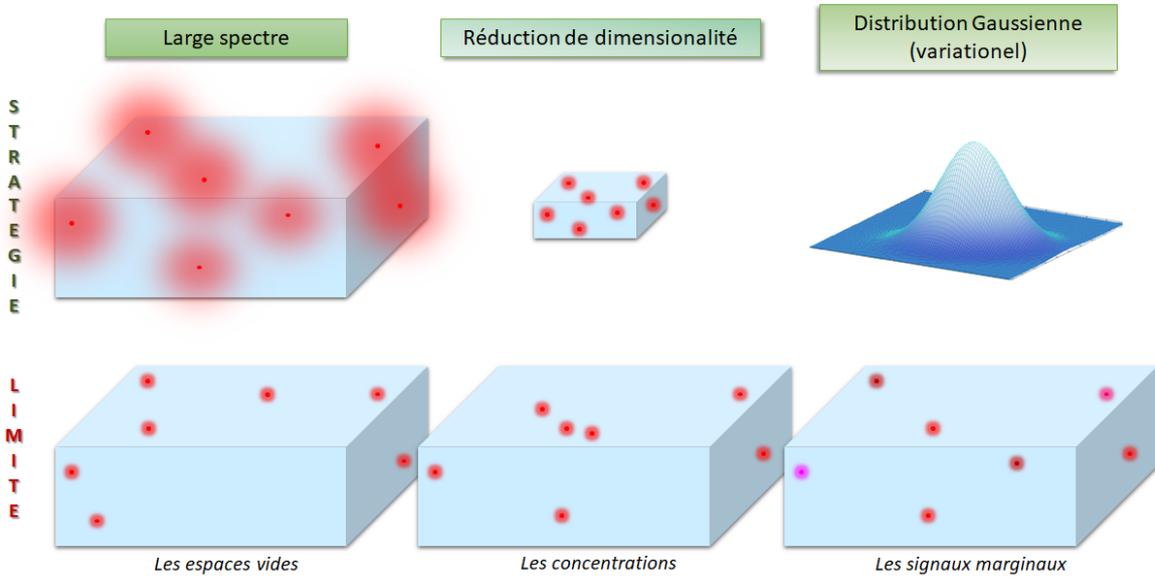


Figure II.7: Exemples de mise en difficulté de chacun des AEs

sortie de son réseau la probabilité qu'à une carte de ne comprendre que des valeurs réelles. On appelle cette information: la vraisemblance. Pour chaque échantillon de carte reconstruite z_s , le discriminateur prédit sa vraisemblance v et sa valeur vient pondérer cette reconstruction dans la moyenne des cartes reconstruites par les AEs. Ainsi les signaux reconstruits spatialement Z_s finaux s'expriment tel que:

$$Z_s(t) = \frac{Z_{sAEDR} V_{AEDR} + Z_{sAEBS} V_{AEBS} + Z_{sVAE} V_{VAE}}{V_{AEDR} + V_{AEBS} + V_{VAE}}(t) \quad (II.1)$$

Où V_{AEDR} , V_{AEBS} et V_{VAE} sont les ensembles des vraisemblances des cartes reconstruites par AE à chaque instant t .

II.1.1.5 Méthode spatio-temporelle

La méthode spatio-temporelle repose sur la combinaison des deux méthodes précédentes pour exploiter les corrélations temporelles intrinsèques à un signal et les corrélations spatiales existantes entre les signaux d'un même ensemble. Deux prédicteurs RNRs, trois AEs RNCs et un discriminateur RNC sont tous impliqués dans chaque itération du traitement point-à-point de la reconstruction spatio-temporelle des signaux comprimés. Elle consiste à appliquer la méthode spatiale à partir de signaux prédits au lieu de signaux comprimés.

Tandis que dans les représentations de données en cartes de la méthode spatiale les valeurs manquantes qu'il faut reconstruire sont identifiées par une valeur constante. Dans la méthode spatio-temporelle, les valeurs manquantes sont supplantés par les valeurs prédites par la méthode temporelle. Il est également possible d'améliorer les prédictions de la méthode temporelle si, à chaque pas de temps, on utilise la reconstruction spatio-temporelle des signaux prédits pour prédire le prochain pas de temps. La reconstruction d'un ensemble de signaux lors d'une itération de reconstruction temporelle peut alors être réalisée telle que l'illustre la Figure II.8.

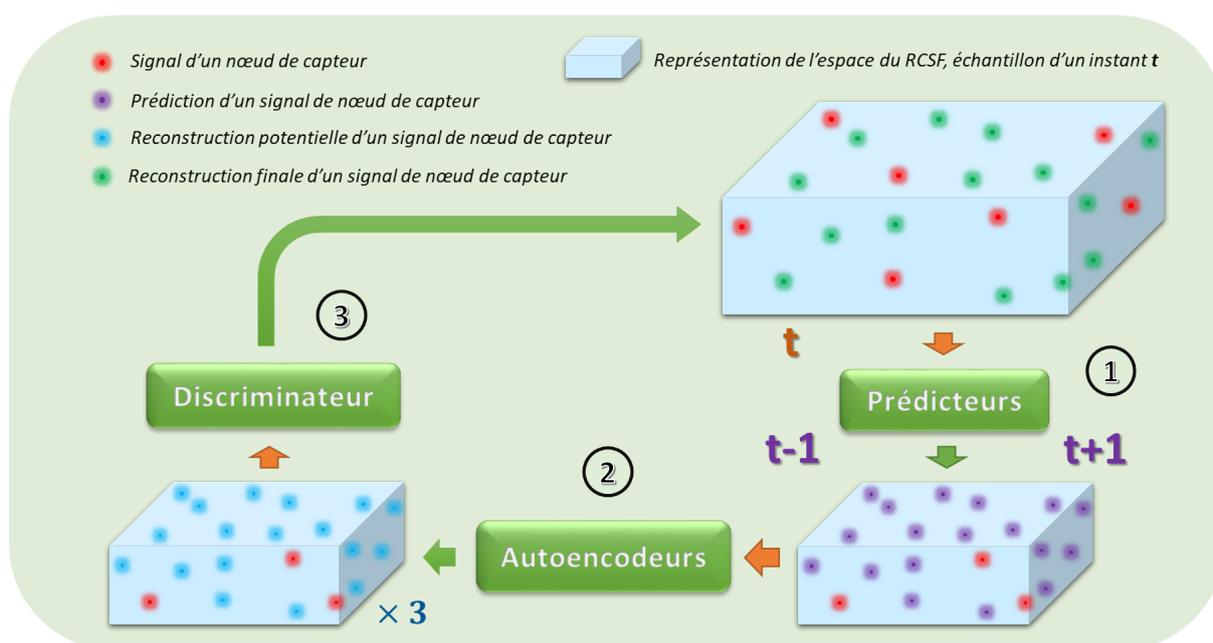


Figure II.8: Schéma d'une itération de reconstruction tempo-spatiale des signaux comprimés d'un RCSF

Pour chaque instant t de l'ensemble des signaux, la première étape de reconstruction est de prédire les valeurs manquantes à l'instant suivant et précédent en utilisant les prédicteurs. Durant la deuxième étape, les cartes constituées de valeurs réelles des signaux comprimées et de valeurs prédites des signaux prédits aux instants $t + 1$ et $t - 1$ sont reconstruites par les AEs, ce qui aboutit à 6 reconstructions de l'ensemble des signaux, une pour chaque AE dans le sens causal et anti-causal de reconstruction. Enfin, les reconstructions des AEs sont moyennées pondéremment à leur score de vraisemblance établit par le discriminateur. Alors pour chaque instant t , la méthode spatio-temporelle établit deux reconstructions aux instant $t + 1$ et $t - 1$, ces dernières sont utilisés comme données d'entrée dans la première étape de la prochaine itération de la méthode. Selon la logique de son ordre de traitement, il convient de l'appeler plus justement la méthode tempo-spatiale. Lorsque que l'ensemble

des signaux est reconstruit ainsi, les reconstructions causale et anti-causale sont moyennées pondéremment à la fiabilité statistique des prédictions tel que c'est décrit dans la méthode temporelle (cf II.1.1.3).

Pour renforcer l'apprentissage des signaux prédits par les AEs, on change la nature de la caractéristique capteur pour qu'elle indique la probabilité qu'une donnée soit issue d'un capteur. Ainsi elle n'est plus binaire, tandis que la valeur 1 signifie toujours que la donnée est acquise par un nœud de capteur et que 0 représente un espace vide, les valeurs flottantes signifient qu'une donnée est originellement manquante et pondèrent l'incertitude représentée par une donnée issue des signaux prédits. Les valeurs de fiabilité statistique des prédictions conviennent parfaitement à numériser cette information.

Cette méthode d'AP tempo-spatiale de reconstruction point-à-point dédiée aux données raster constitue, au stade actuel, l'aboutissement du travail réalisé au cours de cette thèse dans le cadre de son premier contexte d'application. Nos techniques d'apprentissage ensembliste (moyenne pondérée par la fiabilité statistique des prédictions des signaux prédits, moyenne pondérée par le score de vraisemblance du discriminateur des reconstructions spatiales des AEs, l'ajout des prédictions temporelles dans les données spatiales compressées) sont spécifiquement ajustées et dédiées à nos modèles et à nos objectifs.

II.1.2 Les algorithmes de traitement du signal

II.1.2.1 Estimation de la fiabilité statistique d'une prédiction

Algorithm II.1 Calcul des fiabilités statistiques des prédictions

Require: predicted signals ensemble $Z = \{z_{i,j}\}$, complete signals ensemble $X = \{x_{i,j}\}$,
compressed signals ensemble $Y = \{y_{i,j}\}$, margin of error *margin*

Ensure: statistical reliabilities of predictions *reliabilities*

```
l ← length( $Z_0$ )
trials ← [0] × l
successes ← [0] × l
reliabilities ← []
for  $i = 0$  to length( $Z$ )−1 do
     $z$  ←  $Z_i$ 
     $x$  ←  $X_i$ 
     $y$  ←  $Y_i$ 
    start ← 0
    for  $j = 0$  to  $l - 1$  do
        if  $y_j$  is not NULL then
            start ←  $j$ 
        end if
         $k$  ←  $j - \textit{start}$ 
        trials[ $k$ ] += 1
        if abs( $z_j - x_j$ ) < margin then
            successes[ $k$ ] += 1
        end if
    end for
end for
i ← 0
while  $i < l$  and trials $i$  is not 0 do
    reliabilities.insert( $\frac{\textit{successes}_i}{\textit{trials}_i}$ )
     $i$  += 1
end while
return reliabilities
```

La notion de fiabilité statistique représente la probabilité en pourcentage qu'une prédiction établie par un modèle de RNR corresponde exactement avec sa valeur réelle en considérant une marge d'erreur prédéfinie donnée. C'est une grandeur essentielle aux méthodes développées. Elle est utilisée pour pondérer la moyenne qui associe les signaux prédits causalement et anti-causalement, ainsi que pour indiquer la probabilité qu'un signal soit proche d'une mesure de capteur dans le cadre d'une reconstruction spatiale. Cette probabilité a tendance à diminuer à mesure que la distance temporelle qui sépare une prédiction d'une valeur réelle augmente. La fiabilité statistique se cal-

cule par la proportion d'erreur issue de signaux prédits comparativement aux signaux complets originaux pour chaque distance temporelle séparant les valeurs réelles au sein de ces mêmes signaux compressés. Concrètement, il s'agit d'un pourcentage moyen d'exactitude de reconstruction. L'algorithme II.1 permet de le calculer pour chaque horizon de prédiction.

L'exécution de cet algorithme nécessite au préalable que les modèles de prédicteurs aient été entraînés et qu'un ensemble de signaux compressés aient été prédits par ces mêmes modèles.

II.1.2.2 Régularisation d'un signal prédit

Algorithm II.2 Post-traitement d'un signal prédit

Require: predicted signal $z = \{z_i\}$, compressed signal $y = \{y_i\}$, applied tension to reduction $tension = 1$

Ensure: processed predicted signal $z' = \{z'_i\}$

```
 $z' \leftarrow z$   
 $start \leftarrow 0$   
for  $i = 0$  to  $length(z) - 1$  do  
  if  $y_i$  is NULL then  
    continue  
  end if  
  if  $i - start < 2$  then  
     $start \leftarrow i$   
    continue  
  end if  
   $differences \leftarrow []$   
  for  $j = start$  to  $i - 2$  do  
     $differences.insert(z_{j+1} - z_j)$   
  end for  
   $mean_{diff} = mean(differences)$   
   $final_{diff} = y_i - z_{i-1}$   
   $gap = final_{diff} - mean_{diff}$   
  for  $j = i - 1$  to  $start - 1$  do  
     $z'_j += gap \times tension \times \frac{j - start}{i - start}$   
  end for  
   $start \leftarrow i$   
end for  
return  $z'$ 
```

Minimiser la perte de précision de reconstruction à mesure que les prédictions s'éloignent d'une valeur réelle est un enjeu majeur de la méthode temporelle. Afin d'y

contribuer, un algorithme de post-traitement des signaux prédits, donc des signaux reconstruits temporellement, a été développé. L'objectif de ce post-traitement est de tirer un maximum de profit du fait que, contrairement aux travaux de l'état de l'art sur la prévision de données, les prédictions sont au moins en majeure partie comprises dans des intervalles de valeurs réelles. L'idée derrière cet algorithme II.2 est d'apporter de la cohérence et de la régularité au signal reconstruit lorsque la dernière valeur d'une longue série de prédictions se trouve être relativement très éloignée de la valeur réelle qui borne son intervalle de prévision dans son sens de reconstruction. Cela provoque une irrégularité dans la continuité du signal. Si l'on suppose des signaux de nature régulière dans l'application de la méthode temporelle, cet écart signifie que les prédictions ont accumulé des erreurs d'estimation. Pour corriger cette différence tout en conservant la tendance des valeurs prédites, on réduit progressivement l'écart entre les valeurs prédites consécutivement pour tendre vers l'écart moyen μ des valeurs prédites dans l'intervalle des deux valeurs réelles comme illustré sur la Figure II.9.

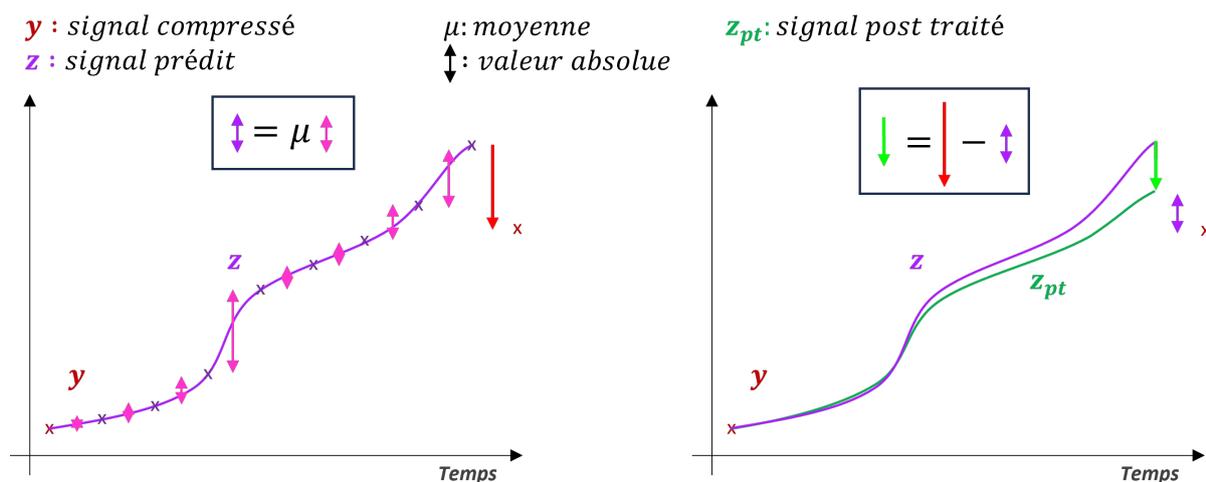


Figure II.9: Illustration du post-traitement d'un signal prédit dans l'intervalle de deux valeurs réelles

Ce traitement n'est intéressant que s'il est appliqué à des signaux que l'on suppose de nature régulière. Lorsqu'on parle de signal post-traité dans la suite de ce chapitre, on parle d'un signal prédit sur lequel on a appliqué cet algorithme II.2. Pour des signaux de nature irrégulière, la section suivante propose un traitement différent.

II.1.2.3 Discrimination des erreurs de prédiction

Nous avons développé un second algorithme de post-traitement des signaux prédits qui se base sur un principe contraire au précédent. Il suppose qu'une discontinuité entre

une valeur réelle et la prédiction qui la précède ne signifie pas nécessairement qu'une imprécision est en jeu. Durant la phase de développement et de tests, nous avons observé que le post-traitement des signaux prédits réduisait considérablement l'erreur de reconstruction d'un signal compressé mais que, ponctuellement, les prédicteurs avaient suivi une bonne tendance du signal que le post-traitement venait amoindrir. Nous avons cherché un moyen de discriminer ce type de situation de celles pour lesquelles l'incertitude avait effectivement fait diminuer la précision. Grâce à un signal de référence sensé généraliser les tendances du signal, on peut discriminer les anomalies contenues dans un signal prédit des valeurs spécifiques au signal complet avec l'algorithme II.3.

Algorithm II.3 Ajustement d'un signal à un signal discriminant

Require: reconstructed signal z , discriminant signal \bar{z} , applied factor to signal amplitude

$margin_factor = 0.1$

Ensure: adjusted signal z'

$z' \leftarrow z$

$amplitude \leftarrow \max(z) - \min(z)$

$margin \leftarrow amplitude * margin_factor$

for $i = 0$ **to** $\text{length}(z) - 1$ **do**

if $\text{abs}(z_i - \bar{z}_i) \geq margin$ **then**

$z'_i = \bar{z}_i$

end if

end for

return z'

Ce traitement considère une marge acceptable de différence entre le signal prédit et le signal discriminant qui borne les valeurs du signal ainsi post-traité. La marge correspond à un pourcentage paramétrable de l'amplitude du signal discriminant. A noter qu'aucun post-traitement des signaux prédits n'est indispensable aux méthodes de reconstructions, ce sont des outils de traitement du signal supplémentaires optionnels qui peuvent améliorer la précision des reconstructions.

II.1.3 Création d'une API dédiée

II.1.3.1 Présentation de deepsignals

deepsignals est une IPA, en langage python, conçue pour permettre l'utilisation des méthodes de reconstructions spatio-temporelles de signaux compressés d'un RCSF développées durant la thèse. Nous avons souhaité que nos méthodes soient généralisables aux données raster, accessibles au plus grand nombre et qu'une application puisse être rapidement implémentée. C'est pourquoi cette IPA propose différents niveaux de configuration et d'utilisation pour s'adapter aux jeux de données et aux besoins des utilisateurs en termes de précision de reconstruction ou de coûts d'exécution. Une attention particulière est portée à la modulation de son architecture pour minimiser les dépendances internes et faciliter son évolution et sa maintenance. Elle est en accès libre sur Gitlab: <https://gitlab.xlim.fr/rousseau/deepsignals>.

deepsignals utilise, entre autres, Keras, Tensorflow, NumPy, pydash et matplotlib. Elle permet d'entraîner, de tester et d'exécuter des RNs pour effectuer des reconstructions spatiales, temporelles et tempo-spatiales à partir d'un jeu de données spatio-temporel raster ou de STs. Elle propose divers outils d'analyse et de rendus graphiques des signaux. Ses dépendances externes directes sont: numpy, keras, tensorflow, matplotlib, pydash, random, math, os, datetime, scipy.

Les sous-sections suivantes présentent, par le biais de ses classes de premier plan, son fonctionnement global. Pour plus de détails, en annexes A.1 se trouve les diagrammes Unified Modeling Language (UML), de : paquetages A.1.1, classes A.1.2, cas d'utilisation A.1.3.

II.1.3.2 La classe Application

La classe Application est la première interface de deepsignals. Elle permet de réaliser rapidement et simplement une application des méthodes deepsignals à un jeu de données raster. Ses fonctions par défaut sont les suivantes :

- Elle initialise un environnement de travail et les réseaux de neurones deepsignals.
- Elle charge les précédentes sessions d'utilisation de deepsignals.
- Elle gère la mémoire vive de la carte graphique de façon autonome.
- Elle propose différents modes d'exécution des méthodes de reconstruction inclu-

ant les phases d'entraînements, de tests et de fonctionnement.

Pour créer une Application, il est nécessaire d'avoir préalablement défini la structure du RCSF ou du système d'où proviennent les données par le biais de la classe `SensorNetworkProfile`, ainsi que les caractéristiques des signaux et des hyperparamètres des réseaux de neurones par le biais de la classe `NeuralNetworkProfile`. Plusieurs options de configuration de l'application sont disponibles à la construction:

- `batch_size`, la taille d'un lot de données (cf I.1.4.5) (32 par défaut)
- `directory`, le dossier dans lequel enregistrer les fichiers de session et de résultats ("application" par défaut)
- `feature_id`, les caractéristiques compressées (0 par défaut)
- `error_margin`, la précision et la fiabilité d'une reconstruction est évaluée à cette marge d'erreur près (0.1 par défaut)
- `reliability`, liste de valeurs de fiabilités statistiques (cf II.1.2.1) (None par défaut, elle sera calculée par l'IPA)
- `test_compression_rates`, liste de taux de compression à tester après l'entraînement des modèles ([0.01, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.95] par défaut)
- `should_regenerate_test_signals`, certains signaux de tests et d'entraînement sont sauvegardés pour accélérer l'utilisation de l'IPA à partir de la seconde session, cette option désactive leur chargement et les écrase si des signaux correspondants sont re-générés (False par défaut)

La classe `Application` n'est pas indispensable mais simplifie grandement l'utilisation de l'IPA.

II.1.3.3 Les classes de profils

Les classes dites de profils sont utiles pour formater des informations relatives à l'application pour automatiser leur traitement avec `deepsignals`. Certaines fonctionnalités requièrent d'avoir préalablement instancié des profils.

La classe `FeatureProfile` définit une caractéristique de signal. Elle est essentielle à l'initialisation d'un modèle de RN `deepsignals` puisqu'elle décrit des propriétés du tenseur de sortie des modèles. Elle comprend un nom, une taille, une fonction

d'activation, une fonction de perte et un poids de perte.

La classe `NeuralNetworkProfile` définit le profil d'un modèle de RN. Elle est essentielle à l'initialisation d'un modèle de réseau de neurones deepsignals. Elle dispose de plusieurs options de construction qui sont: son nom, le chemin de sauvegarde du modèle, l'option de chargement d'un modèle existant, la liste des `FeatureProfile` qui décrivent les entrées et sorties du modèle.

La classe `SensorNetworkProfile` définit une représentation exploitable par les classes deepsignals d'un RCSF. Elle est essentielle à la lecture, la manipulation et l'analyse d'un jeu de données. Elle contient les dimensions relatives (largeur, profondeur, hauteur) de l'espace du RCSF et la liste des descriptions des noeuds de capteurs qui sont des objets, ou dictionnaires, incluant les attributs suivants: un numéro d'identification, une référence (nom), la position dans l'espace relatif du RCSF avec ses coordonnées x,y,z.

II.1.3.4 La classe Dataset

La classe `Dataset` encapsule un jeu de données pour sa lecture et sa modification dans deepsignals. Les classes deepsignals qui ont à faire avec un jeu de données utilisent une instance de la classe `Dataset`. Tous les ensembles de signaux qu'ils soient compressés, complets ou reconstruits sont au format `Dataset`. Une instance peut être sauvegardée au format JSON et ré-instanciée depuis ce format. Cette classe est pourvue d'une trentaine de méthodes qui rendent les données facilement manipulables et lisibles et qui font d'elle la classe la plus volumineuse de l'IPA.

II.1.3.5 Les autres classes

La classe `Numerics` est une classe statique qui produit plusieurs résultats numériques à partir d'instances de `Dataset`. C'est par le biais de ses fonctions que l'on peut observer les différentes performances de reconstruction des signaux.

La classe `Graphics` est une classe statique qui s'appuie sur la classe `Numerics` pour produire des graphes de résultats.

La classe `Compressor` est une classe statique qui masque, autrement dit, qui compresse, les données d'un dataset.

La classe `Reconstructor` est une classe statique qui réalise toutes les différentes méthodes de reconstructions à partir de modèles entraînés et de datasets.

La classe `Environment` détient et configure les informations d'une session d'application `deepsignals`.

Les classes `Training`, `Testing` et `Running` gèrent les différentes phases d'utilisation des RNs et configurent les datasets en fonction.

La classe statique `NeuralNetworkFactory` instancie les modèles de RNs.

La classe `NeuralNetworkModel` est la classe abstraite à la base de tous les modèles de RNs. Les classes `deepsignals` de modèles de RNs encapsule les modèles Keras et Tensorflow en plus des méthodes et attributs spécifiques aux méthodes `deepsignals`.

La classe `Convolutional3DModel` est la classe abstraite parente à tous les modèles de RNC. La classe `Discriminator` ainsi que la classe abstraite `AutoEncoder`, qui est parente à tous les AEs, en héritent. Les classes d'AEs sont `AEDimensionalityReduction`, `AEBroadSpectrum` et `VariationalAE`. Enfin, la classe `Prédicteur` est celle des prédicteurs.

La classe `Reliability` configure, calcule, restitue, charge et sauvegarde des tableaux de fiabilité statistique à partir de datasets.

II.2 Applications

II.2.1 Les ressources d'expérimentation

II.2.1.1 Logicielles et matérielles

Les travaux de thèse ont été exclusivement programmés en Python. Parmi les bibliothèques et IPAs utilisées, on retrouve principalement NumPy qui permet de manipuler et pré-traiter efficacement et rapidement des jeux de données pour des applications d'AP, Keras et Tensorflow. Tensorflow est une plate-forme open source d'AA dédiée à la production. Elle comporte différentes IPAs, pour permettre une grande liberté de développement et de traitement bas niveau en analyse de données, modélisation de RNs et en déploiement de logiciel d'AA en production. Keras est un outil de programmation de haut niveau originellement inventé pour la recherche qui encapsule Tensorflow. Il permet de créer rapidement et facilement des prototypes de RNs. Keras est gratuit d'utilisation, se met continuellement à jour des dernières technologies informatiques et d'IA, et sa communauté est active. Il est notamment utilisé par Google, Netflix, Uber, le CERN, Telp, Square, entre autres. Il est naturellement conçu pour fonctionner de manière transparente sur l'Unité de Traitement Graphique (UTG), ou Graphics Processing Unit (GPU), sans configuration ou démarrage particulier. Ses performances de calculs sont optimisées notamment grâce à des bibliothèques comme Theano, Microsoft Cognitive Toolkit et Cuda. Pour toutes ces raisons, nous avons majoritairement utilisé l'IPA Keras pour concevoir des modèles de RNs ainsi que Tensorflow quand il a été nécessaire de personnaliser ou de créer des couches ou des hyperparamètres spécifiques.

L'ordinateur de travail est pourvu des processeurs suivants: une carte graphique NVIDIA Quadro P5000 et une carte centrale Intel Xeon Silver 4114. La NVIDIA Quadro P5000 a une capacité de mémoire vive de 16 GigaOctets et une bande passante de 288 Go/secondes. Elle possède 2560 cœurs CUDA pour les calculs parallèles et est compatible avec Keras via Cuda. NVIDIA a lancé sa série Pascal en 2016. En 2023, il s'agit d'une carte graphique de milieu de gamme que l'on retrouve à un prix de vente de 2000€.

II.2.1.2 Le réseau de capteur sans fil

Le jeu de données des applications est constitué de mesures de capteurs d'un RCSF installé dans le bâtiment GEII du campus universitaire de Brive-la-Gaillarde. Les noeuds

de capteurs sont répartis sur les trois étages et le rez-de-chaussée du bâtiment. Ils mesurent les paramètres d'ambiance de température et d'Humidité Relative (HR). Le réseau communique avec Zigbee qui est un protocole de communication destiné aux réseaux sans fil à faible débit (250 kbits/s), faible portée et faible consommation pour un moindre coût. La topologie du réseau en étoile convient bien à ce type de protocole. Les noeuds du réseau sont des terminaux qui peuvent seulement communiquer avec la station de base. Elle est équipée d'une base de données locale qui est transmise à un serveur de données et réinitialisée une fois par semaine. C'est un routeur Linux Meshlium (Figure II.10) qui communique à la fois sur le RCSF et sur un réseau Ethernet connecté au serveur.



Figure II.10: Station de base Meshlium



Figure II.11: Noeud de capteurs Wasmote

Les noeuds de capteurs sont des Wasmote (Figure II.11) qui sont spécialement conçus pour les applications à faible consommation et optimisés pour les modes de fonctionnement en veille. Le Wasmote est un dispositif électronique terminal équipé d'un micro-contrôleur qui permet l'intégration de nombreux modules tel que de la mémoire, des capteurs et des communicateurs. Il autorise la communication avec des protocoles sans fils comme le Zigbee, le Wifi, le Bluetooth, le LoRaWAN et la 3G, pour en citer quelques uns.

Le capteur utilisé est un DHT22 thermo-hydrique (Figure II.12) qui réalise une mesure en 2 secondes, ce qui est rapide en considérant les autres capteurs de température



Figure II.12: Capteur de température et d'humidité relative DHT22

Alimentation	Consommation	Plage de mesure	Marge d'erreur	Dimensions
3,3~6Vcc	1mA~1,5mA 50 μ A en veille	0 à 100% HR -40 à +80°C	\pm 2% HR \pm 0,5°C	25 x 15 x 9 mm

Table II.1: Caractéristiques du capteur DHT22

et d'humidité. Cela participe également à la durée de vie du RCSF. Le tableau II.1 donne les caractéristiques d'utilisation de ce capteur.

II.2.1.3 Le jeu de données

Nous avons expérimenté les méthodes de reconstructions sur des signaux de températures. Ils ont été mesurés du 18 février 2020 au 15 janvier 2021 avec une interruption du 1^{er} novembre au 14 décembre. La particularité de cet ensemble est qu'il est lui même sous-échantillonné pour être reconstruit par AC. Chaque mesure a été relevée selon une matrice de détection aléatoire pour une période de réveil des capteurs constante de 30 minutes dans un réseau où les horloges des noeuds sont synchronisées, comme détaillé dans la thèse de Frank Itoua-Engoti [9]. Ce qui implique que les signaux sont compressés à des taux légèrement différents. Par exemple, la Figure II.13 montre les signaux acquis par les noeuds de capteurs 002 et 007. On observe qu'un premier signal regroupe 10770 mesures tandis que le second en totalise 10935 dans le même intervalle de temps. La matrice de mesure est équiprobable selon le rapport global $\rho = \frac{n}{m}$ (pour rappel: n la dimension du signal, m le nombre de mesures échantillonnées). Dans le cas présent: $\rho = 1,4$ et $n = 457776$. Tous les signaux qui constituent le jeu de données sont affichés en annexes A.2.1.

La première problématique des applications a été de reconstituer un jeu de donnée

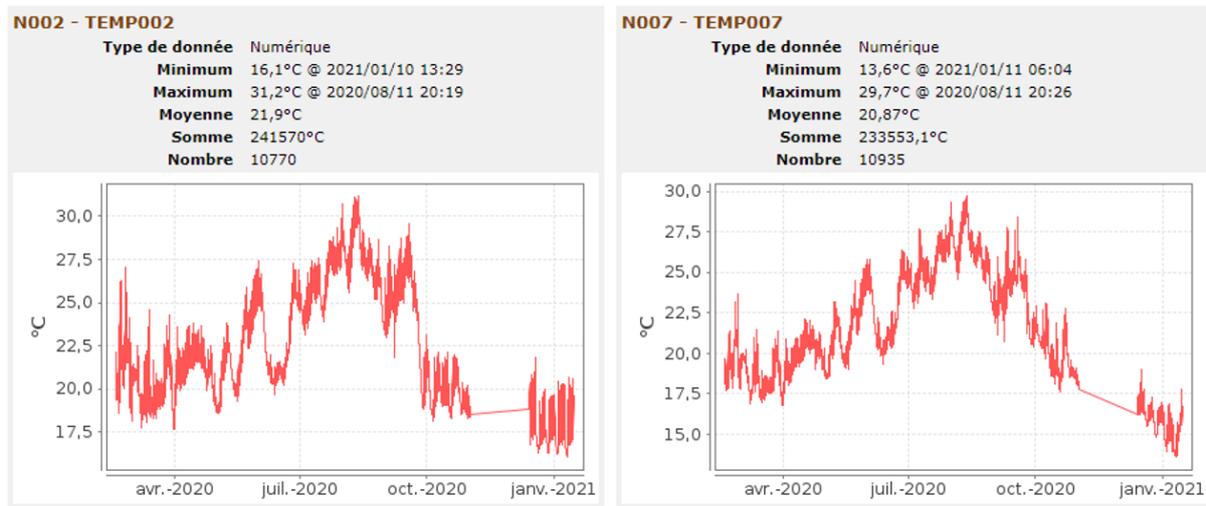


Figure II.13: Exemple de signaux de températures des noeuds de capteurs 002 et 007 du rez-de-chaussée du bâtiment GEII de Brive-la-Gaillarde

complet à partir de celui-ci. C’est-à-dire, un jeu de données comprenant des signaux complets de même longueur selon le même pas de temps. Il doit préserver les corrélations temporelles des signaux et les corrélations spatiales du RCSF pour servir à l’entraînement de tous les RNs engagés dans les expérimentations. Nous avons procédé de deux façons.

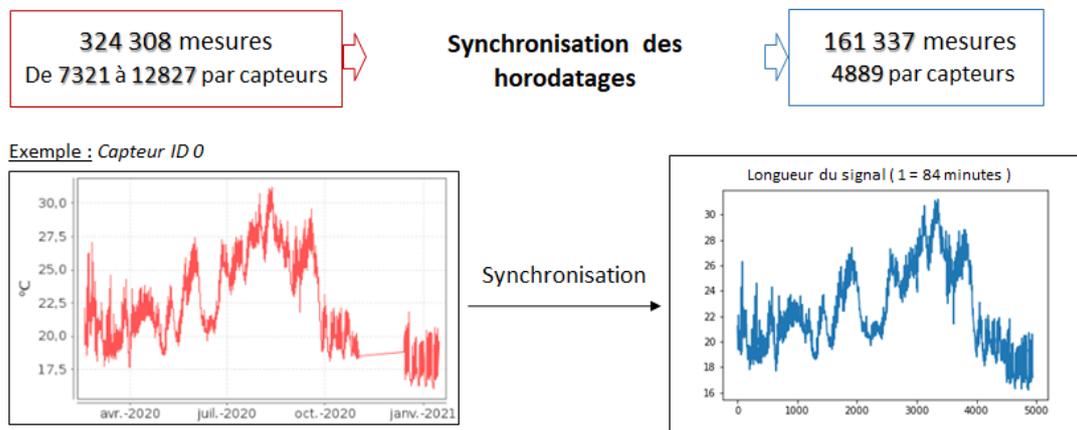


Figure II.14: Synchronisation des horodatages des signaux du RCSF

L’application de la méthode temporelle a été réalisée à partir d’un jeu de donnée de 33 signaux de 4889 valeurs de températures échantillonnées à un pas de temps de 84 minutes. Ces signaux ont été obtenus à partir des originaux en réduisant la longueur de tous les signaux à la longueur maximale pour laquelle chaque signal connaît une mesure de température comprise sur l’échelle d’un même pas de temps. On appelle cette opération la synchronisation des horodatages, illustrée Figure II.14. Ainsi le jeu

de données d'entraînement des modèles de prédicteur n'est constitué que de valeurs réelles mais il est également relativement restreint, ce qui ne convient pas à la seconde application, celle de la méthode tempo-spatiale.

La longueur des signaux correspond au nombre d'échantillons disponibles dans une représentation spatiale (cf II.1.1.2), or 4889 échantillons ne sont pas suffisants pour entraîner, valider et tester efficacement les RNCs. Lors des premières expérimentations de la méthode temporelle, nous avons comparé ces résultats de reconstruction avec une méthode intuitive simple: l'interpolation polynomiale spline de degré 3. Nous avons observé qu'une simple interpolation de ce genre convenait relativement à combler les données manquantes d'un signal compressé lorsque le taux de compression descendait en dessous de 85% en considérant la marge d'erreur du capteur DHT22 de 0,5°C, voir Figure II.15.

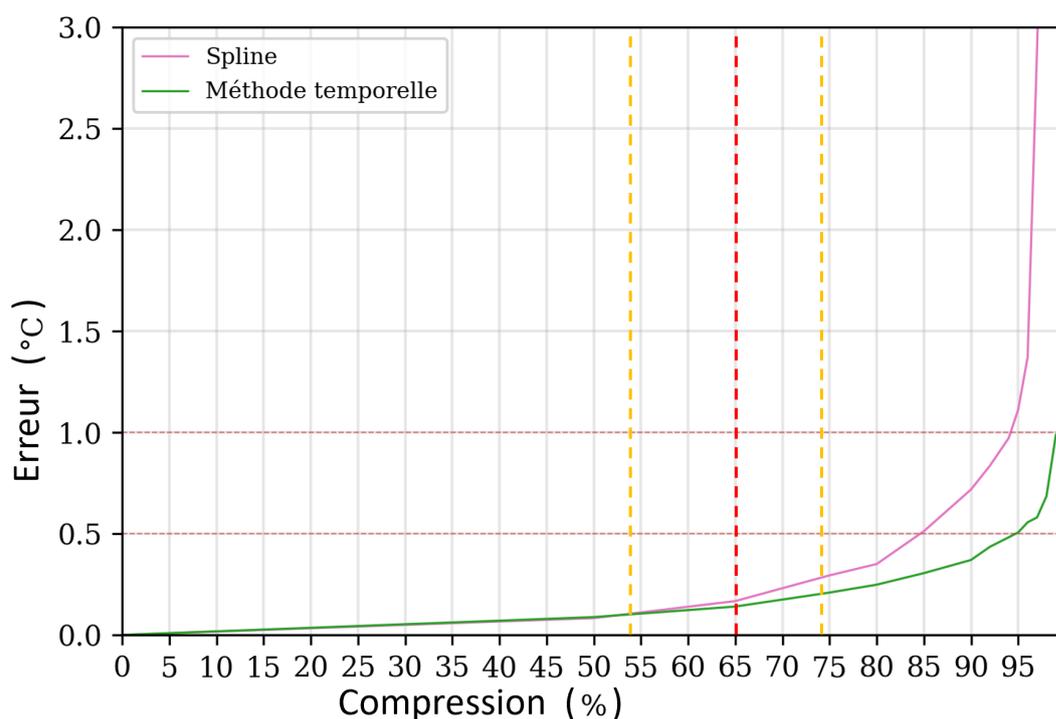


Figure II.15: Erreur moyenne de reconstruction des signaux de températures par interpolation spline en fonction du taux de compression

Le jeu de données d'origine comptabilise 324308 mesures de températures réparties irrégulièrement sur les 33 signaux, avec un minimum de 7321 et un maximum de 12827 mesures par signal. En respectant un pas de temps constant de 15 minutes

entre chaque valeur de température pour chacun des signaux aux dates auxquelles les relevés ont été effectués, le nombre de mesures serait de 916014, soit 27758 valeurs par signal. Au final, les signaux réellement mesurés correspondent à une version compressée à 65% de ce que serait ces signaux pour un pas de temps d'acquisition de 15 minutes, avec un minimum de 54% et un maximum de 74% de taux de compression par signal. Au regard des résultats observés sur la Figure II.15, on sait qu'à ces grandeurs de compression la méthode d'interpolation spline présente relativement peu d'erreurs de reconstruction de nos signaux. L'interpolation des valeurs manquantes pour un pas de temps de 15 minutes a permis d'augmenter le nombre d'échantillons de données à 27758 au lieu de 4889 pour l'application des méthodes de reconstruction spatiale et spatio-temporelle, comme l'illustre la Figure II.16.

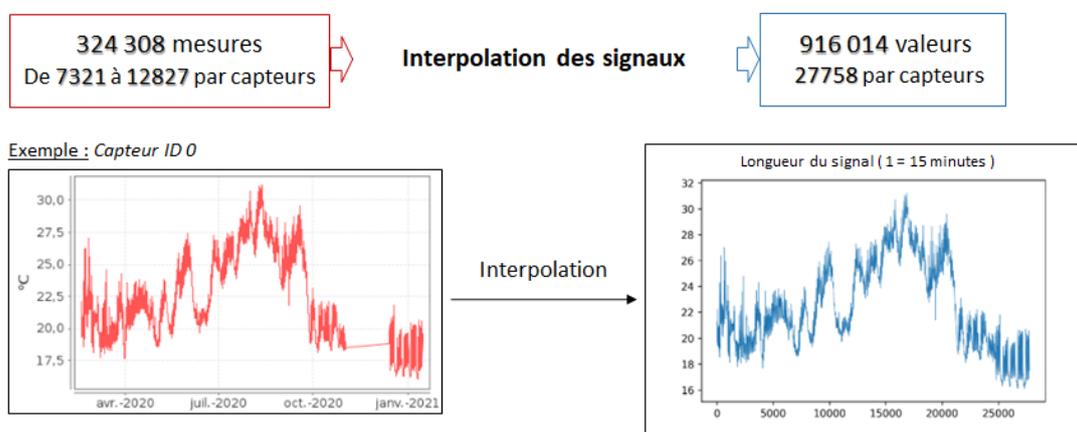


Figure II.16: Interpolation des valeurs des signaux du RCSF

A chaque fois, on tronque la période du 1^{er} novembre au 14 décembre durant laquelle le RCSF était éteint.

II.2.2 Application de la méthode temporelle

II.2.2.1 Le modèle et l'entraînement des prédicteurs

La Figure II.17 illustre le modèle de prédicteur pour cette application.

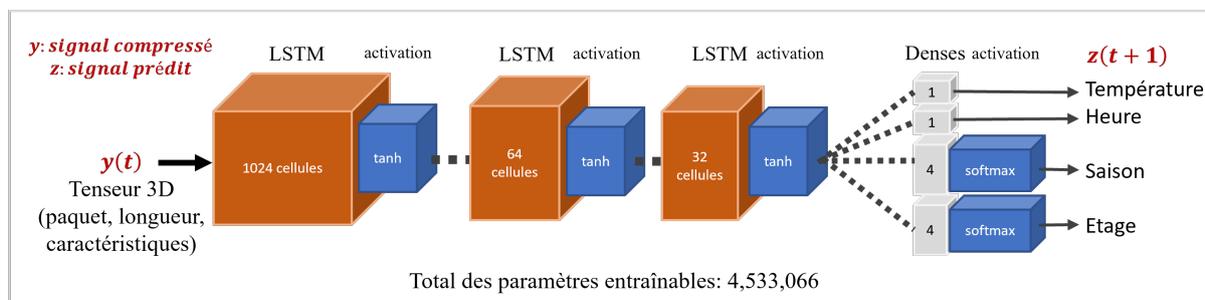


Figure II.17: Modèle de prédicteur utilisé pour l'application de la méthode temporelle

Il est composé de trois couches récurrentes de cellules LSTMs et de quatre couches de sorties qui correspondent aux caractéristiques des signaux, soit la température, l'heure de l'acquisition, la saison et l'étage du bâtiment. Ces caractéristiques aident le RN à comprendre les tendances et courbes des signaux de température. Les sorties de saison et d'étage sont activées par une fonction softmax pour classer le signal parmi quatre possibilités qui sont, respectivement: l'été, l'automne, l'hiver, le printemps et le rez-de-chaussée, le premier, le deuxième, le troisième étage. Les autres sorties ne sont pas activées pour conserver leur propriété numérique. Les données de températures d'apprentissage sont standardisées (distribution uniforme de moyenne nulle avec un écart-type de 1). La fonction d'activation tangente hyperbolique s'adapte bien à cette normalisation des signaux en produisant des sorties bornées entre -1 et 1. L'entraînement est configuré avec un optimiseur RMSprop pour effectuer une descente de gradients stochastique en mettant à jour la moyenne mobile des carrés des gradients, ce qui permet d'ajuster le facteur d'apprentissage au cours des époques. Les fonctions de perte sont une entropie croisée catégorique (Categorical Cross Entropy) pour le problème de classification de la saison et l'étage et une erreur moyenne absolue (MAE) pour le problème de régression de l'heure et la température. Le choix de ces hyperparamètres découlent de nombreux essais d'optimisation de l'apprentissage durant lesquelles les hyperparamètres ont beaucoup variés. Ils sont adaptés aux signaux étudiés et ont donné les meilleurs résultats.

Le jeu de données d'entraînement est composé de 33 signaux issus de 33 noeuds de capteurs, 23 de ces signaux sont utilisés pour entraîner les prédicteurs, 9 servent à la validation tandis qu'un est utilisé pour les tests. Le tenseur d'entraînement d'entrée est

en 3 dimensions, il est constitué de 112424 échantillons, il y en a 4888 issus de chaque signal. Chacun de ces échantillons comprend un seul point, donc une séquence de longueur 1, d'un signal caractérisé par 10 valeurs (1 de température, 1 pour l'heure, 4 pour les saisons, 4 pour les étages) qui correspondent aux caractéristiques de sortie du modèle. Le tenseur d'entraînement de sortie fait exactement les mêmes dimensions sauf que les valeurs des signaux qu'il contient correspondent aux valeurs consécutives des valeurs du tenseur d'entrée. C'est cela qui permet à un modèle de prédicteur d'apprendre à prédire les valeurs suivantes en partant des précédentes. Un des deux prédicteurs apprend le mécanisme inverse en inversant l'ordre de ces tenseurs.

II.2.2.2 Les résultats

Pour cette application, l'algorithme II.1 a estimé les fiabilités statistiques du tableau II.2 en considérant une marge d'erreur de 0,5°C qui correspond à l'incertitude de mesure des capteurs DHT22 du RCSF. Alors, on considère qu'une valeur de température reconstruite qui présente une différence inférieure à 0,5°C avec sa valeur réelle est correctement estimée. Ces pourcentages de fiabilité statistique ont servi à pondérer la moyenne des signaux prédits et post-traités.

Prédiction	1	2	3	4	5	6	7
Fiabilité	87%	76%	66%	56%	50%	45%	40%
Prédiction	8	9	10	11	12	13	14
Fiabilité	38%	36%	35%	33%	33%	30%	29%
Prédiction	15	16	17	18	19	20	21+
Fiabilité	28%	27%	25%	26%	26%	25%	25%

Table II.2: Fiabilité statistique pour différents horizons de prédictions de l'application (exprimés en nombre de pas de temps)

Afin d'évaluer les performances de la méthode, nous l'avons comparée à l'état-de-l'art en implémentant l'ACK [10], l'E²GAN-RF [84] et un AE bidirectionnel [92], [99], noté BAE, basé sur des couches LSTMs. Dans un souci de rigueur de notre approche comparative, nous avons également travaillé à développer une version alternative du modèle E²GAN-RF, qui est originellement conçu pour apprendre à partir de jeux de données incomplets qu'il faut imputer, pour qu'il exploite pleinement les informations contenues dans le jeu de données d'entraînement complet de l'application. Nous avons

profité que l'architecture de son générateur soit sensiblement proche d'un AE pour lui ajouter un pré-entraînement en lui demandant de générer les signaux complets à partir de la partie encodée des signaux compressés avant qu'ils soient observés par le discriminateur. Ainsi, le générateur apprend à tromper le discriminateur tout en présentant un biais de génération qui l'induit à la reconstruction. On appelle ce modèle l'E²GAN-CRF pour Complete Real-data Forcing. La méthode d'ACK est la seule à utiliser des corrélations spatiales pour établir ses reconstructions. Elle est incluse dans le lot des méthodes comparatives pour confronter en premier lieu les méthodes d'AP avec l'AC. Enfin, afin de démontrer par la pratique qu'il est important d'impliquer deux RNRs dans la méthode temporelle, on expose également les résultats de reconstruction que l'on obtient avec un signal prédit. Il est noté en tant que méthode LSTM par la suite.

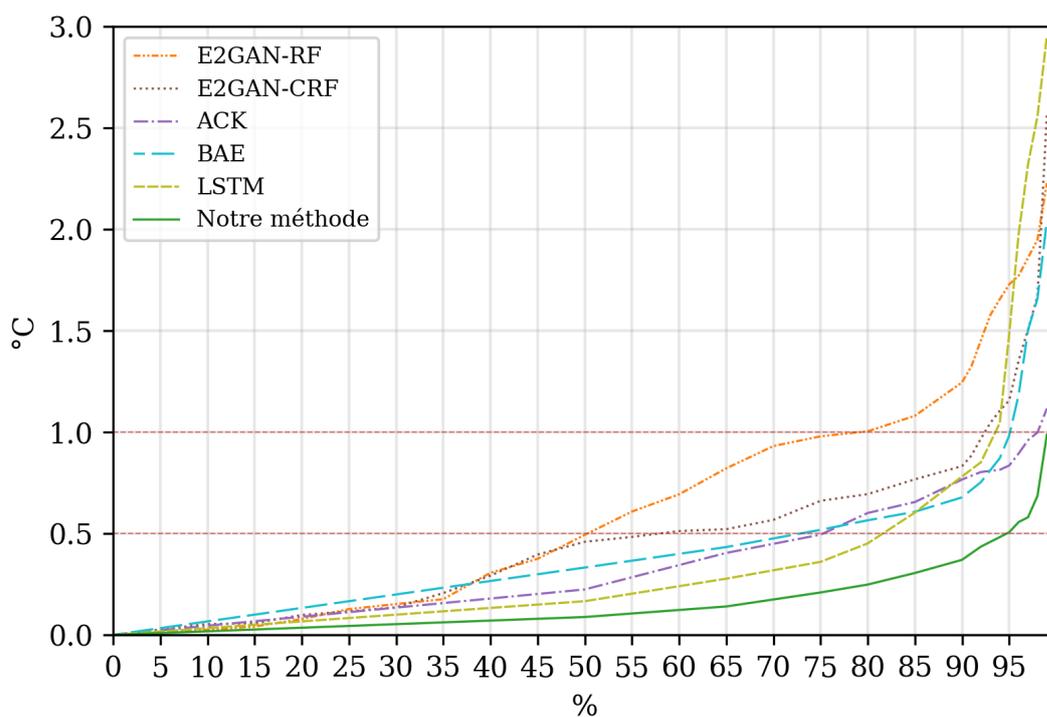


Figure II.18: Erreur moyenne de reconstruction de signaux de températures par taux de compression

La Figure II.18 montre les courbes d'erreurs moyenne de reconstruction de signaux de température issus du jeu de données de l'application en fonction du taux de compression pour chaque méthode de reconstruction. Les RNs engagés dans les reconstructions n'ont pas été entraînés à partir des signaux qui ont servis à établir ces résultats pour ne pas les fausser. On peut faire les observations suivantes:

- la méthode temporelle développée ici est la plus précise, et ce, pour tous les taux de compression.
- un RNR seul comme celui nommé LSTM est efficace pour résoudre des problèmes simples, constatant sa précision pour des taux de compression inférieurs à 85%,
- des modèles complexes comme un BAE ou l'E²GAN-CRF obtiennent de bons résultats pour des taux de compression élevés en ayant une compréhension plus large du jeu de données d'entraînement.
- la méthode par ACK démontre sa précision et sa fiabilité constante à travers chaque niveau de compression, ce qui est attendu étant donné que c'est la seule méthode comparative qui exploite les inter-corrélations entre les signaux d'un même ensemble.

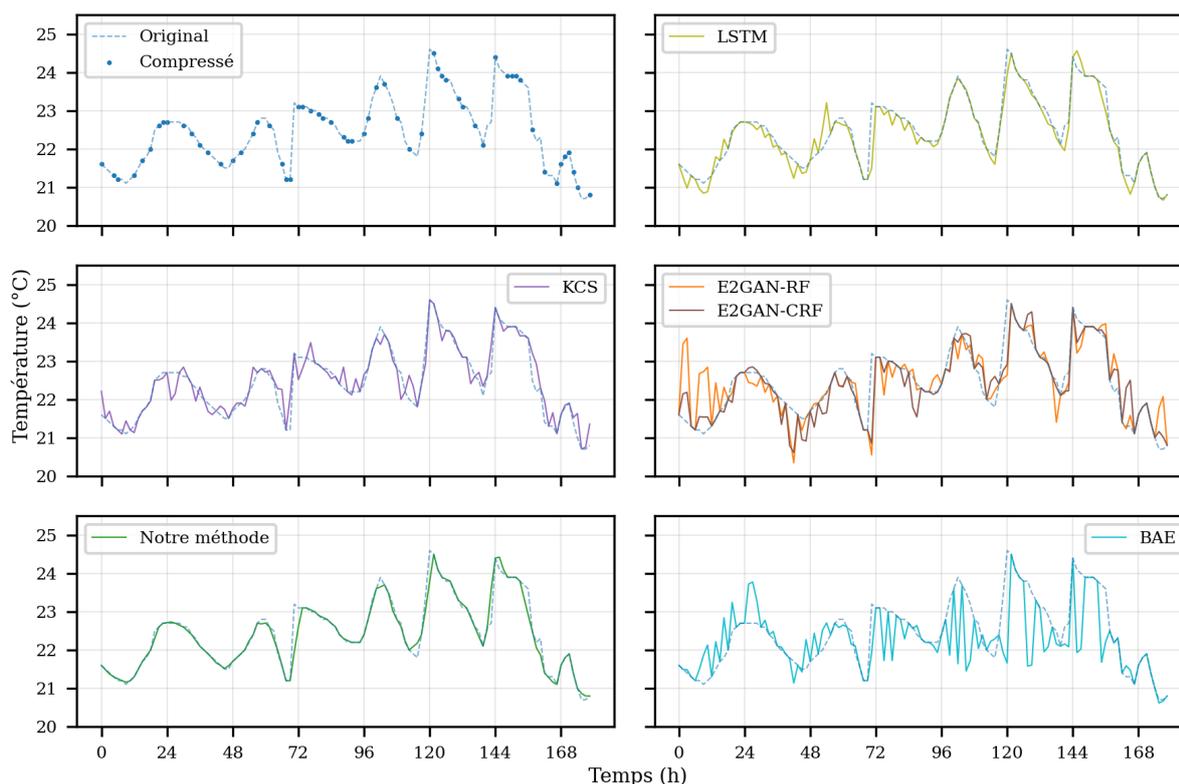


Figure II.19: Exemple de reconstruction d'un signal test comprimé à 50%

Les Figures II.19, II.20 et II.21 montrent des exemples de reconstruction pour les différentes méthodes comparatives et à différents niveaux de compression. Les signaux complets sont toujours sous-échantillonnés aléatoirement, cela impacte fortement la non-répétabilité des résultats pour les méthodes LSTM, BAE, E²GAN-RF et E²GAN-

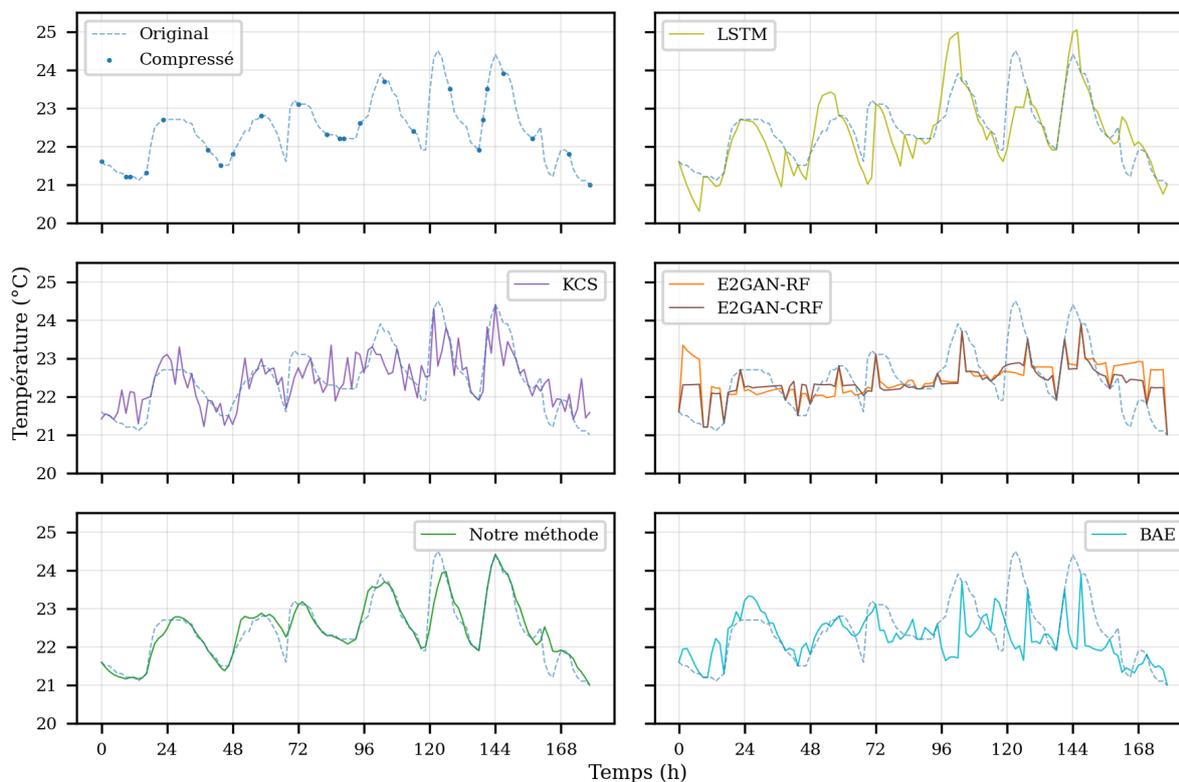


Figure II.20: Exemple de reconstruction d'un signal test comprimé à 80%

CRF lorsque les taux de compression sont très élevés. Ce phénomène apparaît lorsque deux valeurs réelles d'un signal comprimé sont très éloignées l'une de l'autre. Pour un même taux de compression, les signaux comprimés à partir d'un même signal complet peuvent être très différents les uns des autres. Dans la Figure II.21 on observe trois points consécutifs qui apportent peu d'informations supplémentaires chacun. Si un ou deux de ces points avaient été échantillonnés à un autre moment, certaines des reconstructions données en exemples auraient pu s'avérer plus précises. Les expérimentations et la Figure II.18 ont montrés que notre méthode reste fiable et présente de faibles variations dans ses erreurs de reconstruction à des taux de compression équivalents.

Chaque méthode met plus ou moins de temps à calculer une reconstruction. Pour mettre en lumière les différences de temps de calculs qui existent, le tableau II.3 expose des temps de calculs pour une reconstruction de l'ensemble du jeu de données de l'application. La méthode de la thèse n'est pas des plus rapides, cela est principalement dû au fait que c'est une méthode de calcul point-à-point contrairement au BAE et à l'E²GAN-RF qui traitent un signal dans toute sa longueur en une seule fois. L'avantage

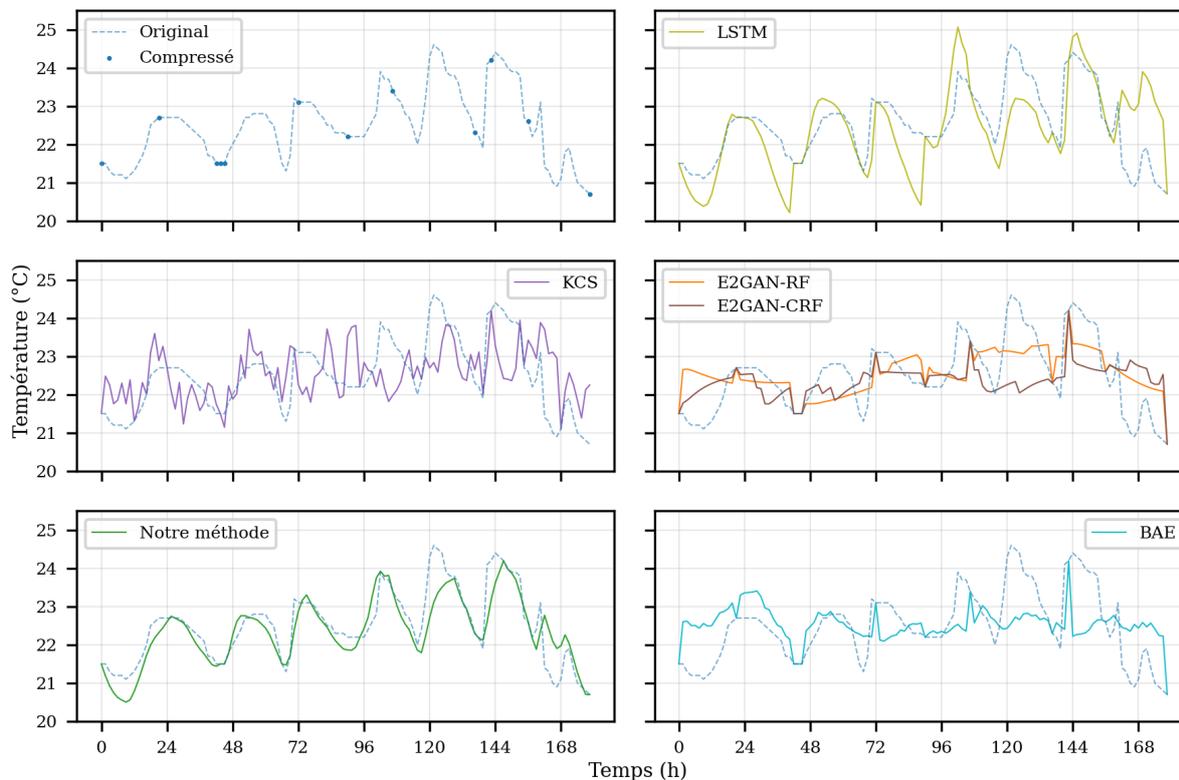


Figure II.21: Exemple de reconstruction d'un signal test comprimé à 90%

Taux de compression	90%	80%	50%
LSTM	396s	347s	230s
BAE	127.15s	127.15s	125.63s
ACK	6291s	8667s	11878s
E ² GAN-RF	22.726s	20.499s	22.338s
E ² GAN-CRF	10.352s	10.567s	10.138s
Notre méthode	819s	743s	512s

Table II.3: Temps de calculs de reconstruction du jeu de données entier d'un période de 9 mois (33 signaux de longueur 4889)

qu'à l'ACK à exploiter les inter-corrélations entre les signaux lui doit aussi l'inconvénient d'être extrêmement coûteux à s'exécuter. Remarquablement, la méthode de la thèse se montre plus précise que celle de l'ACK tout en étant beaucoup plus rapide.

Concrètement, si l'on considère un cas d'usage dans lequel on souhaite reconstruire des

signaux compressés à 90% avec notre méthode toutes les 24 heures et que l'on veut observer les valeurs de ces signaux toutes les 15 minutes. Il faudrait à peu près 0.49 secondes pour achever la reconstruction d'un signal avec notre matériel (cf II.2.1.1). Dans le cas du RCSF de cette application (33 signaux) cela prendrait donc 16.08 secondes une fois par jour. Du point de vue du coût énergétique que représenterait ce cas d'usage, un noeud de notre RCSF consommant 5.835 W pour réaliser l'acquisition et la transmission d'une mesure de température (cf tableau II.4), des signaux compressés à 90% pour un pas de temps de 15 minutes auraient un coût de 1.85 kW/jour.

Mesure	Conversion	Transmission	Microprocesseur	Station	UTG
4 mW	700 mW	63 mW	68 mW	5 W	160 W

Table II.4: Consommation énergétique du matériel de cette application

En incluant l'utilisation de l'UTG pour réaliser la reconstruction des signaux compressés, le coût total journalier serait de 4.42 kW tandis que l'acquisition de ces mêmes signaux complets aurait un coût de 18.5 kW. Pour ce cas d'usage, le gain énergétique que représente l'utilisation de la méthode spatio-temporelle est un facteur 4. Or, il faudra compter 11 jours pour que l'utilisation de la méthode de reconstruction rembourse le coût énergétique de 192 kW nécessaire à l'entraînement des RNs, qui prend une vingtaine de minutes pour se faire dans cette application. (Ce nombre de jour correspond au rapport entre le coût de l'entraînement et le coût journalier d'acquisition des signaux complets.) Ces chiffres omettent tous les coûts liés au stockage des données qui seraient, de toute manière, à l'avantage d'une méthode de compression et de reconstruction qui répond fondamentalement à des problématiques de coûts de stockage de données.

II.2.3 Application de la méthode tempo-spatiale

II.2.3.1 Les modèles

Pour cette seconde application, le modèle de prédicteur, illustré Figure II.22, est basé sur celui de la première. Son rôle ne change pas, il effectue toujours la même reconstruction à base de prédictions.

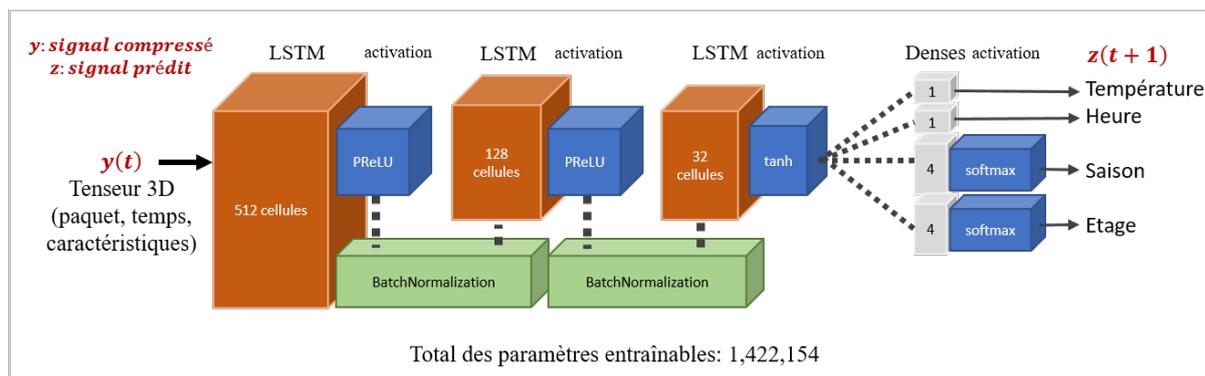


Figure II.22: Modèle de prédicteur utilisé pour l'application de la méthode tempo-spatiale

Des couches de normalisation par lot, ou batch normalization [100], améliorent l'apprentissage tout en permettant de réduire le nombre de cellules. Ces couches stabilisent la distribution de chaque entrée des couches internes du RN pour réduire le décalage de covariance des données. La normalisation est effectuée indépendamment pour chaque caractéristique pour obtenir une moyenne nulle et une variance de 1 par lot de données (cf I.1.4.5). Ce procédé a plusieurs avantages qui facilitent l'entraînement: il le rend plus rapide, il diminue les risques de voir apparaître l'explosion ou la disparition des gradients et réduit le sur-ajustement en participant à généraliser le problème que le modèle doit résoudre. Ces couches de normalisation ont permis d'alléger le prédicteur pour des performances équivalentes, voire meilleures.

Tous les modèles de cette application utilisent des fonctions d'activation Parametric Rectified Linear Unit (PReLU) après les premières couches de leur réseau. Cette fonction d'activation est une variante de la fonction ReLU, qui est très populaire. La fonction ReLU filtre efficacement les informations non déterminantes à la résolution du problème pour une grande majorité de tâches et d'applications or, se faisant, elle a tendance à éliminer définitivement certaines informations qui pourraient être utiles à quelques échantillons de données particuliers à traiter. La fonction PReLU intègre un paramètre entraînable qui détermine l'angle plus ou moins aigu responsable de l'élimination des informations peu utiles au modèle. Sa flexibilité permet de sauveg-

arder des informations à priori négligeables et s'adapte très bien à des méthodes qui souhaitent généraliser au maximum leur modèle.

Les modèles d'AEs sont majoritairement composés de couches convolutives 3D conçues pour assimiler les données représentées dans un espace en 3 dimensions comme c'est le cas d'un RCSF. Les autres couches sont spécifiques à chaque stratégie de reconstruction spatiale, ainsi que les hyperparamètres. Les sorties correspondent avec les caractéristiques du jeu de données, donc avec les sorties du modèle de prédicteur également à l'exception de l'étage qu'il n'est plus besoin de caractériser étant donné que cette information est spatiale et directement représentée dans le format de la donnée. Une caractéristique supplémentaire de sortie de ces modèles est la caractéristique capteur (cf II.1.1.2) qui permet d'identifier les positions de l'espace qui comprennent un signal de noeud de capteurs. Elle peut également recevoir des valeurs de fiabilité statistique de prédiction conformément à la méthodologie (cf II.1.1.5). La figure II.23 illustre le modèle à réduction de dimensionalité, la figure II.24 celui à large spectre et la figure II.25 le modèle variationnel. Pour la suite du manuscrit, ces AEs sont respectivement notés AEDR, AEBS et VAE en se basant sur leur acronyme anglais.

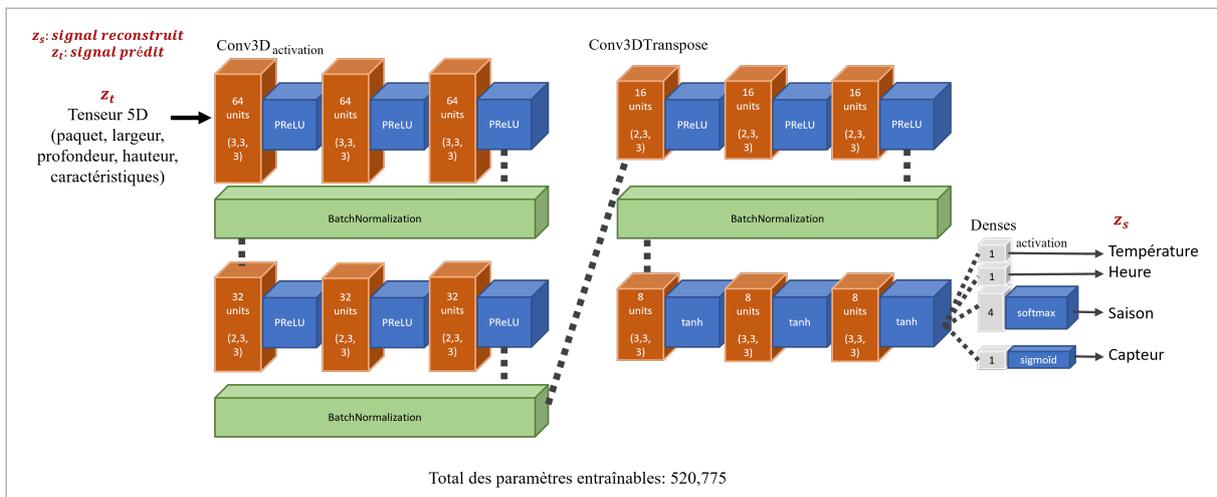


Figure II.23: Modèle d'AE à réduction de dimensionalité pour cette application (AEDR)

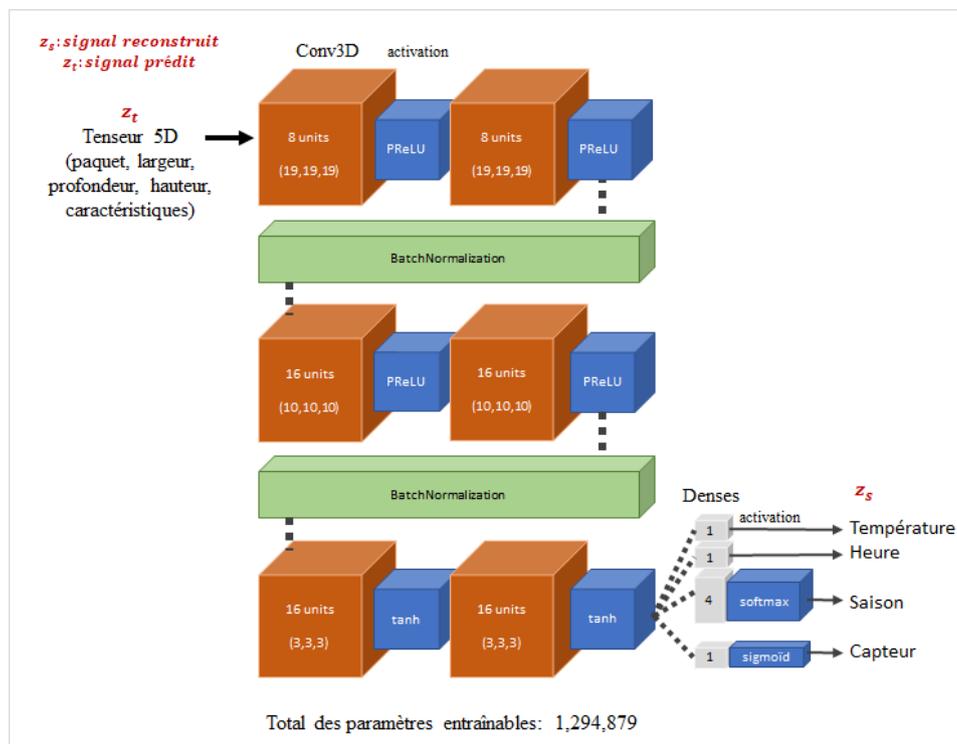


Figure II.24: Modèle d'AE à large spectre pour cette application (AEBS)

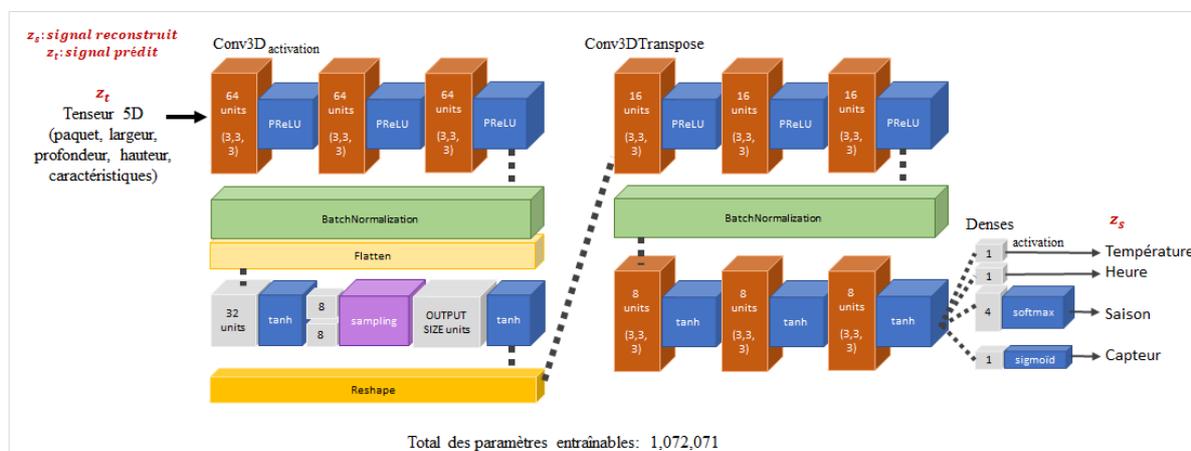


Figure II.25: Modèle d'AE variationnel pour cette application (VAE)

Le discriminateur est relativement simple, voir Figure II.26. Il possède peu de paramètres entraînaibles et des couches de Dropout pour généraliser au maximum ses prédictions. Le dropout, qui ne se traduit pas bien en français, diminue le sur-ajustement en désactivant aléatoirement des neurones différents à chaque rétropropagation. Le RN doit alors réussir à résoudre le problème en ne comptant uniquement que sur les neurones activés. Cette perturbation a pour effet de désynchroniser un tant soit peu les neurones mais exige du réseau que son modèle soit plus général. Utiliser à tort trop de dropout

peut rendre l'apprentissage impossible. Pour un problème de classification binaire tel que celui du discriminateur qui doit déterminer si une carte de données est réelle ou reconstruite, la fonction de perte utilisée est souvent l'entropie croisée binaire. Or, associée à une activation sigmoïd, cette fonction a pour particularité de tendre vers l'une des classes sans se soucier de la probabilité qu'à la donnée de correspondre à l'autre classe, ce mode de résolution se veut sans ambiguïté et très déterministe. La fonction de perte de sortie du discriminateur est une entropie croisée catégorique éparsée, ou sparse categorical cross entropy, associée à une fonction d'activation softmax pour permettre au discriminateur de conserver une distribution égale à 1 entre les deux sorties possibles afin d'éviter que les valeurs de vraisemblance qu'il prédit soient absolument binaires, donc 1 ou 0.

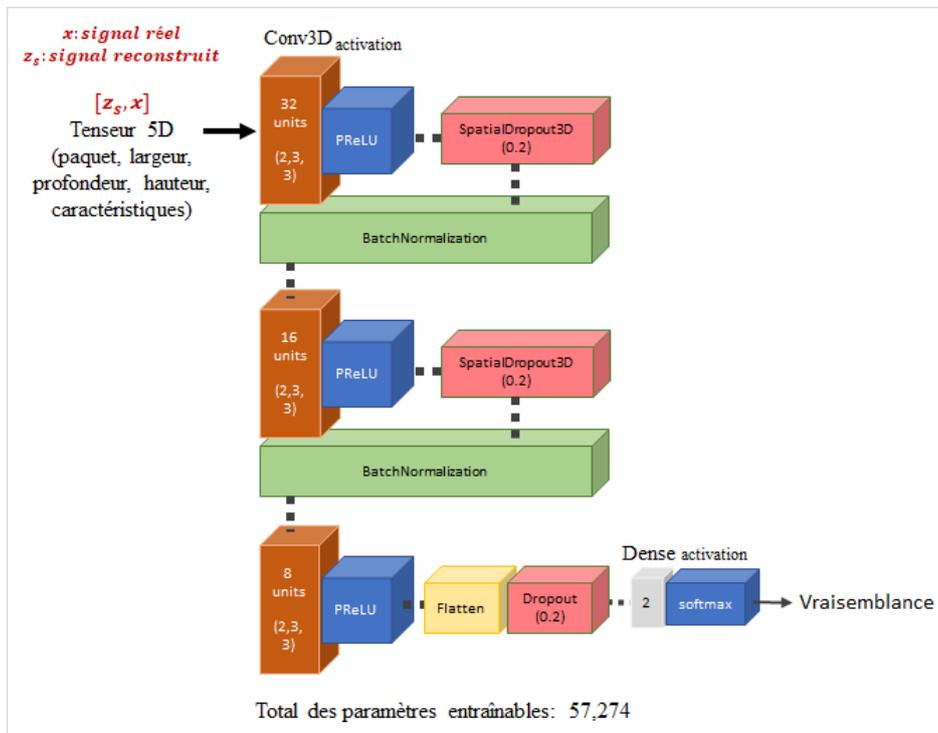


Figure II.26: Modèle de discriminateur pour cette application

Ce modèle particulier de discriminateur réalise l'objectif qui est de déterminer la probabilité qu'à une carte reconstruite d'être réelle en se basant sur l'hypothèse que la précision de reconstruction est corrélée avec cette probabilité. Durant les expérimentations, on a observé qu'un modèle de discriminateur plus complexe, plus performant et respectant les fonctions de sortie standard d'un problème de classification binaire décorréait l'erreur de reconstruction du caractère réaliste de la donnée. C'est-à-dire qu'un discriminateur performant parvient autant, avec une précision équivalente, à

identifier une carte reconstruite qui présente peu d'erreur de reconstruction qu'une carte qui présente beaucoup d'erreurs. On souhaite que le discriminateur soit assez performant pour reconnaître une carte reconstruite mais pas assez performant pour qu'il conserve une incertitude lorsque une reconstruction est relativement précise. Cela permet de corrélérer l'erreur de reconstruction à la vraisemblance.

II.2.3.2 L'entraînement

L'entraînement spatial nécessite simultanément les 33 signaux issus du RCSF pour apprendre les corrélations spatiales. Le tenseur d'entraînement est en 5 dimensions, il est constitué de 27758 échantillons, chaque échantillon est une carte du RCSF à un instant du temps comme expliqué en détails dans la section II.1.1.2. On définit une unité de distance u qui correspond à la distance entre les deux noeuds de capteurs les plus proches dans le réseau. Selon cette unité u , les dimensions de l'espace du RCSF de cette application sont de 24 de largeur, 16 de profondeur et 4 de hauteur. Chaque point de cet espace contient 7 valeurs (température, heure, été, automne, hiver, printemps, capteur) qui correspondent aux caractéristiques de sortie des modèles d'AEs.

Pour permettre aux AEs d'apprendre suffisamment de cas de figures pour être performant de manière générale, on utilise deux jeux de données compressés: un dans lequel chaque signal est compressé aléatoirement entre 10 et 99% de façon équiprobable et un autre dans lequel chaque signal est compressé à 99% temporellement puis spatialement. Chacun de ces jeux de données est reconstruit temporellement par les deux prédicteurs, résultant en quatre jeux de données, un par mode de compression et par prédicteur. Le nombre d'échantillons ainsi obtenus est de 111032, 77722 servent à l'entraînement, les autres à la validation des modèles.

Le taux d'apprentissage est moitié moins élevé pour les AEs, qui traitent des données plus volumineuses, que pour les prédicteurs. Il est ajusté pour éviter au maximum que l'apprentissage se stabilise sur un minimum local pendant la descente de gradients.

Le jeu de données d'entraînement du discriminateur est constitué d'une proportion équivalente du jeu d'entraînement complet d'origine et de cartes reconstruites par les AEs, soit 27758 échantillons réels et 27758 échantillons reconstruits. Les échantillons reconstruits sont aléatoirement tirés dans les reconstructions de chaque AE des 111032 échantillons précédemment cités. Soit un total de 333096 cartes parmi lesquelles 27758 sont utilisées pour entraîner le discriminateur. La répartition de ce tirage est équitable pour chaque AE.

Les valeurs de fiabilité statistique des prédictions (calculées avec l’algorithme II.1) pour cette application sont indiquées dans le tableau II.5. Elles sont plus nombreuses par rapport à la première application puisque les signaux sont plus longs, le nombre de prédictions à imputer dans l’intervalle de deux valeurs réelles est forcément plus élevés pour de très hauts taux de compression. Ce qui signifie également que les variations entre deux valeurs de température sont plus basses étant donné un pas de temps plus court. C’est pourquoi la marge d’erreur utilisée est de 0,2°C pour cette application. De plus, la marge d’erreur doit être adaptée à une distribution uniforme des valeurs de fiabilité statistique autant que la nature des signaux puisse le permettre. La reconstruction tempo-spatiale est favorisée si la caractéristique de capteur embarquée par les AEs est uniformément répartis entre 0 et 1 pour informer distinctement du niveau d’incertitude des valeurs des signaux prédits.

Prédiction	1	2	3	4	5	6
Fiabilité	99,75%	98,63%	97%	95%	92,63%	89,93%
7	8	9	10	11	12	13
86,81%	82,74%	78,18%	73,38%	68,78%	63,81%	58,92%
14	15	16	17	18	19	20
54,67%	50,67%	46,81%	43,24%	40,08%	37,55%	35,59%
21	22	23	24	25	26	27
32,38%	29,54%	28,44%	28,16%	28,41%	26,31%	25,66%
28	29	30	31	32	33	34
23,40%	21,27%	20,75%	23,25%	24,81%	23,68%	21%
35	36	37	38	39	40	41
18,52%	18,46%	19,23%	19,56%	14,70%	17,24%	10%
42	43	44	45	46+		
12,50%	16,67%	18,18%	11,11%	10%		

Table II.5: Fiabilité statistique pour différents horizons de prédictions de l’application (exprimés en nombre de pas de temps)

II.2.3.3 Les résultats

La Figure II.27 montre un exemple de reconstructions spatiales. L'axe des abscisses est l'identifiant d'un noeud de capteurs allant de 0 à 32, les points du graphe indiquent la valeur de température donnée par chaque capteur à un même instant t .

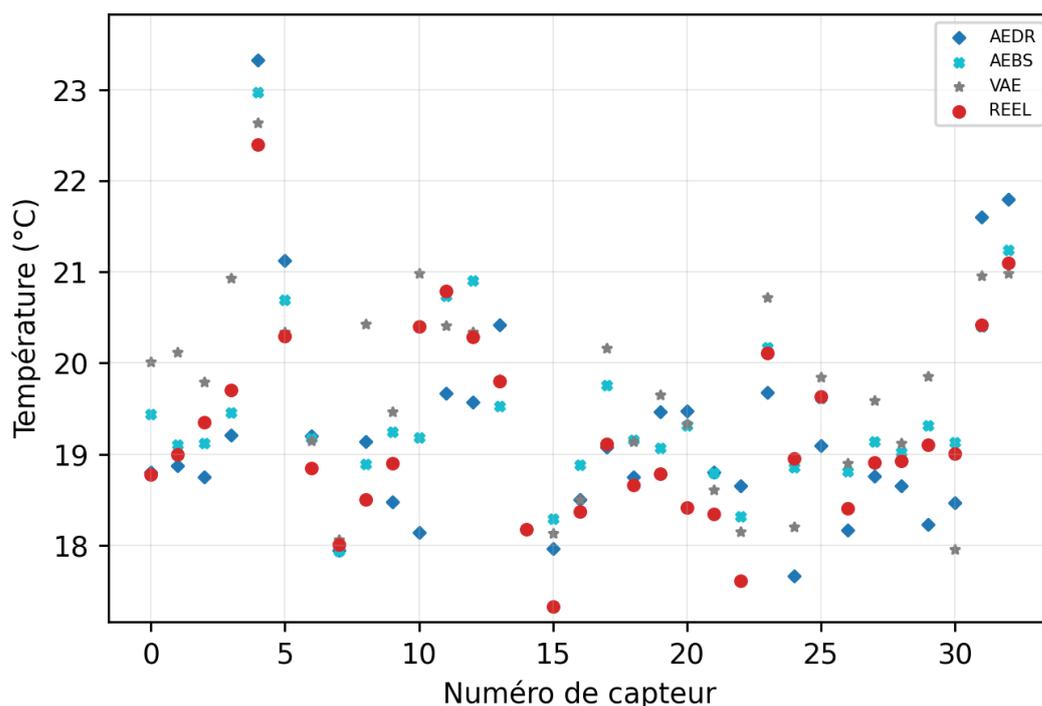


Figure II.27: Exemple de reconstructions spatiales à un instant t par: autoencodeur à réduction de dimensionalité (AEDR), autoencodeur à large spectre (AEBS), autoencodeur variationnel (VAE)

Les points ronds et rouges sont les valeurs réelles, ce sont des mesures acquises à l'instant t par tous les noeuds de capteurs du RCSF. Les autres points ont été générés par les différents modèles entraînés d'AEs à partir des signaux comprimés. On observe que chaque AE estime différemment les valeurs des signaux réels. Dans cet exemple, l'AE variationnel est le plus précis pour le capteur n°4; pour le n°8 c'est celui à large spectre et pour le n°18 c'est celui à réduction de dimensionalité. Le capteur n°14 n'affiche qu'un seul point rond et rouge, cela signifie que cette valeur est comprise dans la carte compressée et n'a pas besoin d'être reconstruite. Pour chaque carte compressée il existe un AE plus précis qu'un autre pour la reconstruire. En moyenne, l'AE à large spectre, qui est aussi le plus complexe, s'en sort mieux que les autres, comme illustré sur la Figure II.28. Cependant, la précision supplémentaire remarquable

qu’apporte les moyennes, pondérées ou non, des reconstructions justifie amplement l’utilisation de trois AEs. Pour autant, si une application nécessite d’optimiser la rapidité de traitement et les coûts, un seul AE peut assumer la partie spatiale de la méthode de reconstruction tempo-spatiale. Pour une application de ce genre, au regard des résultats et de leur architecture, le VAE ou l’AEDR seront les meilleurs choix pour ce rôle.

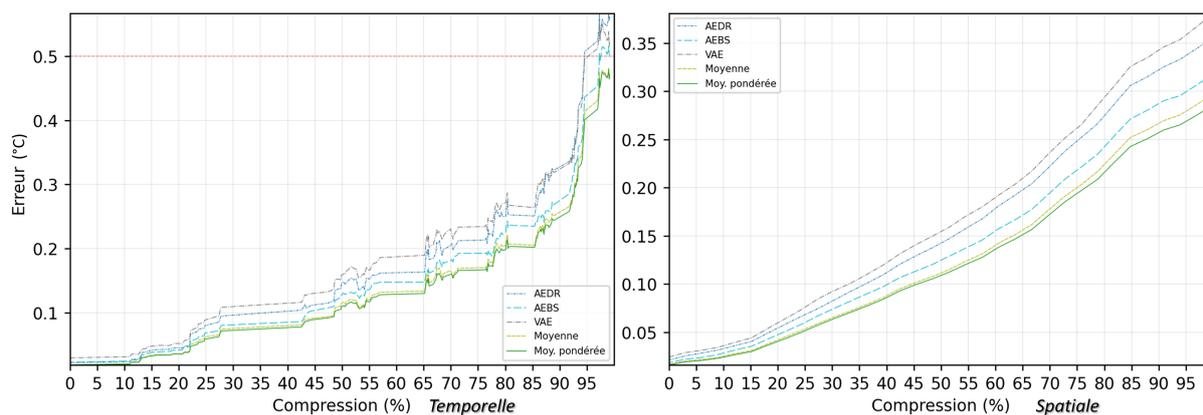


Figure II.28: Erreur moyenne de reconstructions spatiales de signaux de températures par taux de compression

La moyenne pondérée par la vraisemblance résulte en la meilleure précision de reconstruction spatiale. Cela s’explique par la réussite du discriminateur à établir une corrélation entre l’erreur de reconstruction et l’aspect réaliste d’une carte de donnée. Cela se produit si le discriminateur peut lui aussi identifier les corrélations spatiales au sein de la donnée afin de déceler les moindres petites incohérences dans les données reconstruites. La Figure II.29 met en confrontation la vraisemblance des reconstructions avec leur précision. Bien que présentant un effondrement de l’erreur en dessous de 25% de vraisemblance, cette courbe est la meilleure que nous obtenons pour notre application. Cela prouve que le résultat qu’on attend du discriminateur est compliqué à atteindre et que l’apprentissage de cette corrélation est particulièrement sensible.

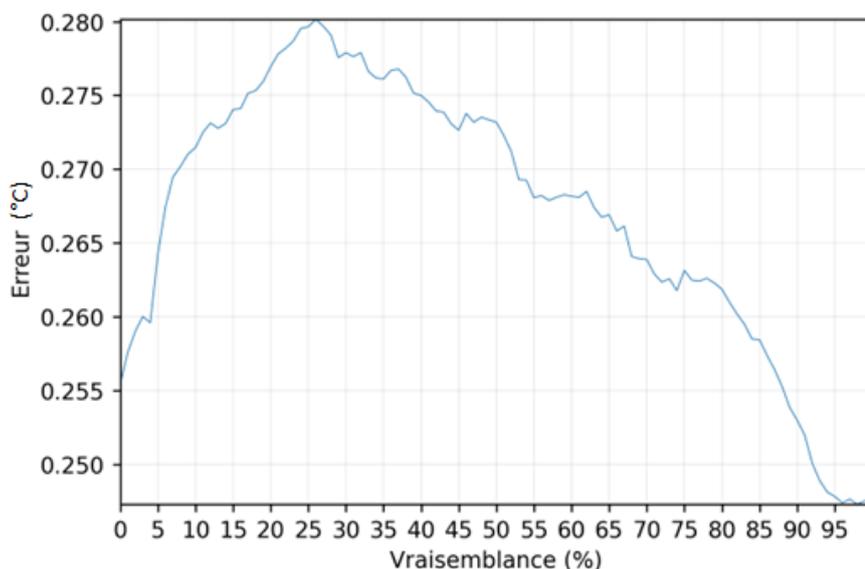


Figure II.29: Erreur moyenne de reconstruction de signaux de températures par vraisemblance

L'apprentissage des corrélations spatiales a permis d'améliorer la précision des signaux prédits. La Figure II.30 montre que l'erreur moyenne de reconstruction a diminué grâce à la méthode spatiale.

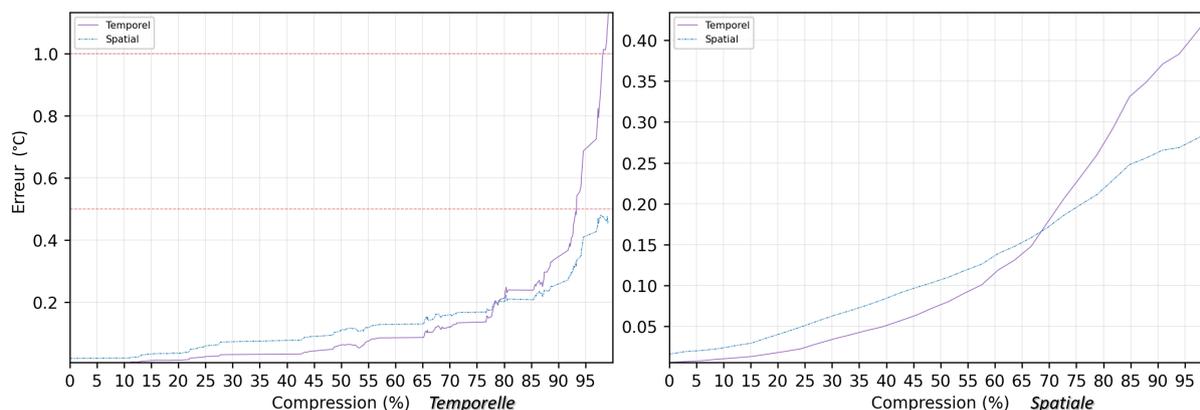


Figure II.30: Comparaison des erreurs moyenne de reconstructions spatiale et temporelle de signaux de températures par taux de compression

Néanmoins, on observe que pour des faibles taux de compression, la transformation des signaux prédits par les AEs a introduit de l'erreur dans les valeurs de températures. Ce n'est pas le problème qu'il y paraît car en réalité cela est simplement la conséquence d'une forte précision dans les premières prédictions des RNRs. Directement à la suite

d'une valeur réelle, une valeur prédite présente peu d'incertitude. En traçant les courbes d'erreur moyenne temporelle et spatiale en fonction de la fiabilité statistique, comme sur la Figure II.31, on peut confirmer les performances de la méthode temporelle dans ces cas spécifiques.

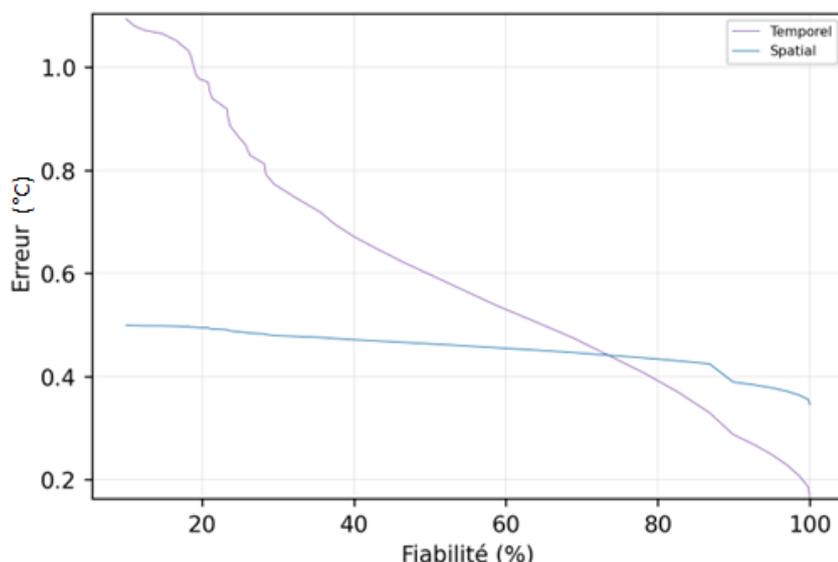


Figure II.31: Erreur moyenne de reconstruction de signaux de températures par fiabilité statistique de prédiction

La méthode tempo-spatiale conserve les valeurs des signaux prédits lorsque leur fiabilité statistique est supérieure à 70%. La Figure II.32 montre les erreurs moyenne de reconstruction des signaux de températures en fonction du taux de compression en incluant des signaux post-traités pour montrer les résultats sur l'ensemble des travaux.

En guise de démonstration, les Figures II.33, II.34, II.35, II.36, II.37, II.38 montrent des exemples de reconstruction de signaux hautement comprimés d'une semaine de températures relevées toutes les 15 minutes, ce qui donne des signaux de longueur 672. Chacun de ces exemples provient de noeuds de capteurs différents, tirés au hasard, à des périodes de l'année différentes. La méthode tempo-spatiale surpasse en précision la méthode temporelle.

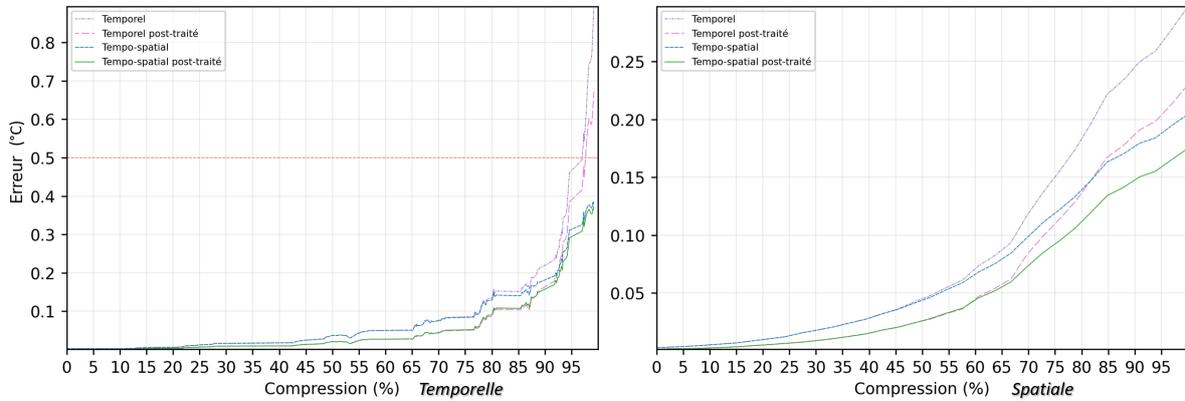


Figure II.32: Erreur moyenne de reconstructions de signaux de températures par taux de compression

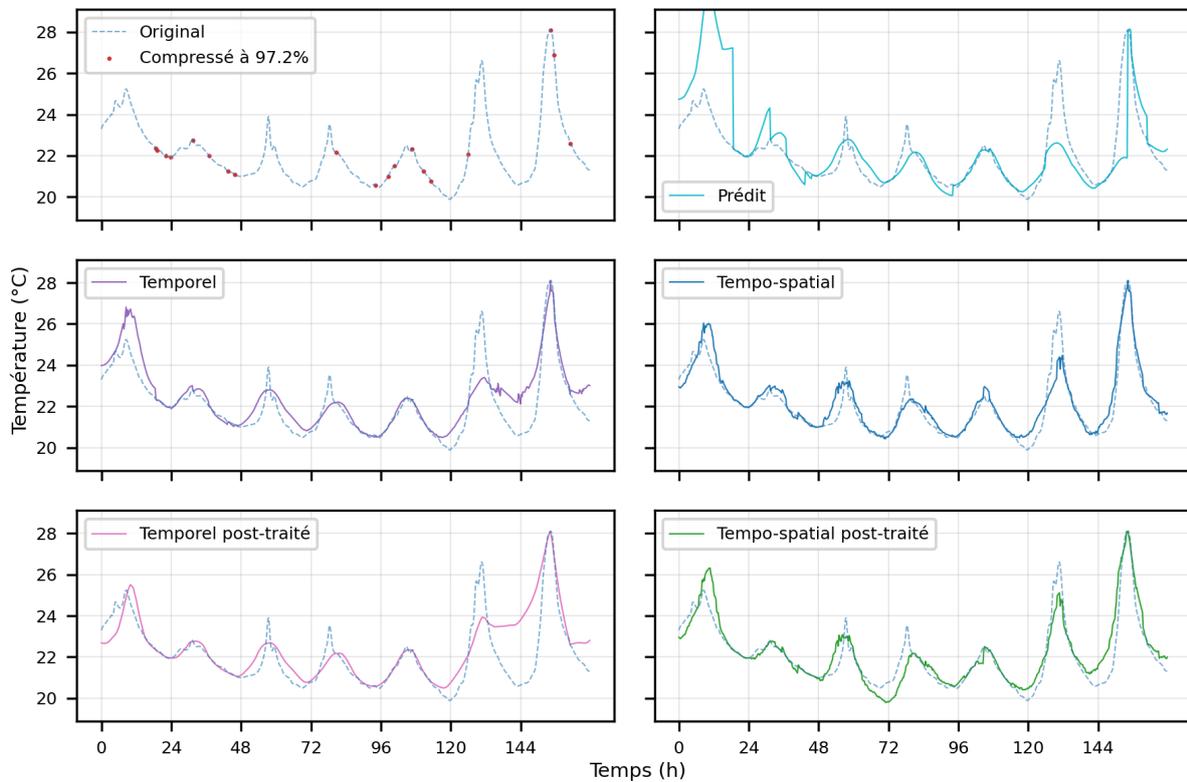


Figure II.33: Exemple de reconstruction d'un signal test compressé à 97,2%

Chapitre II – Développement de méthodes de reconstruction des signaux comprimés par apprentissage profond – application au réseau de capteurs

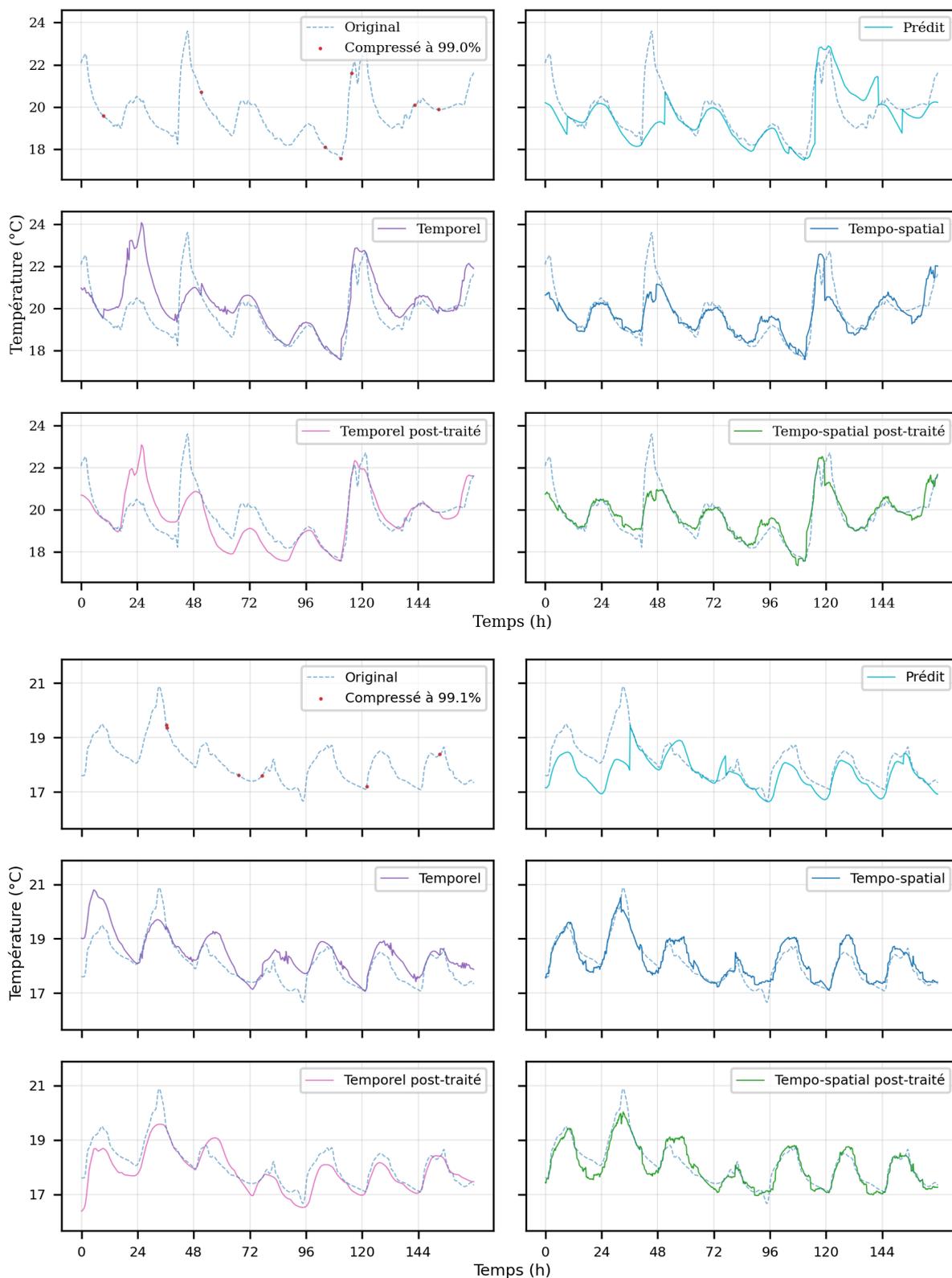


Figure II.34: Exemple de reconstructions de signaux tests comprimés à 99% et 99,1%

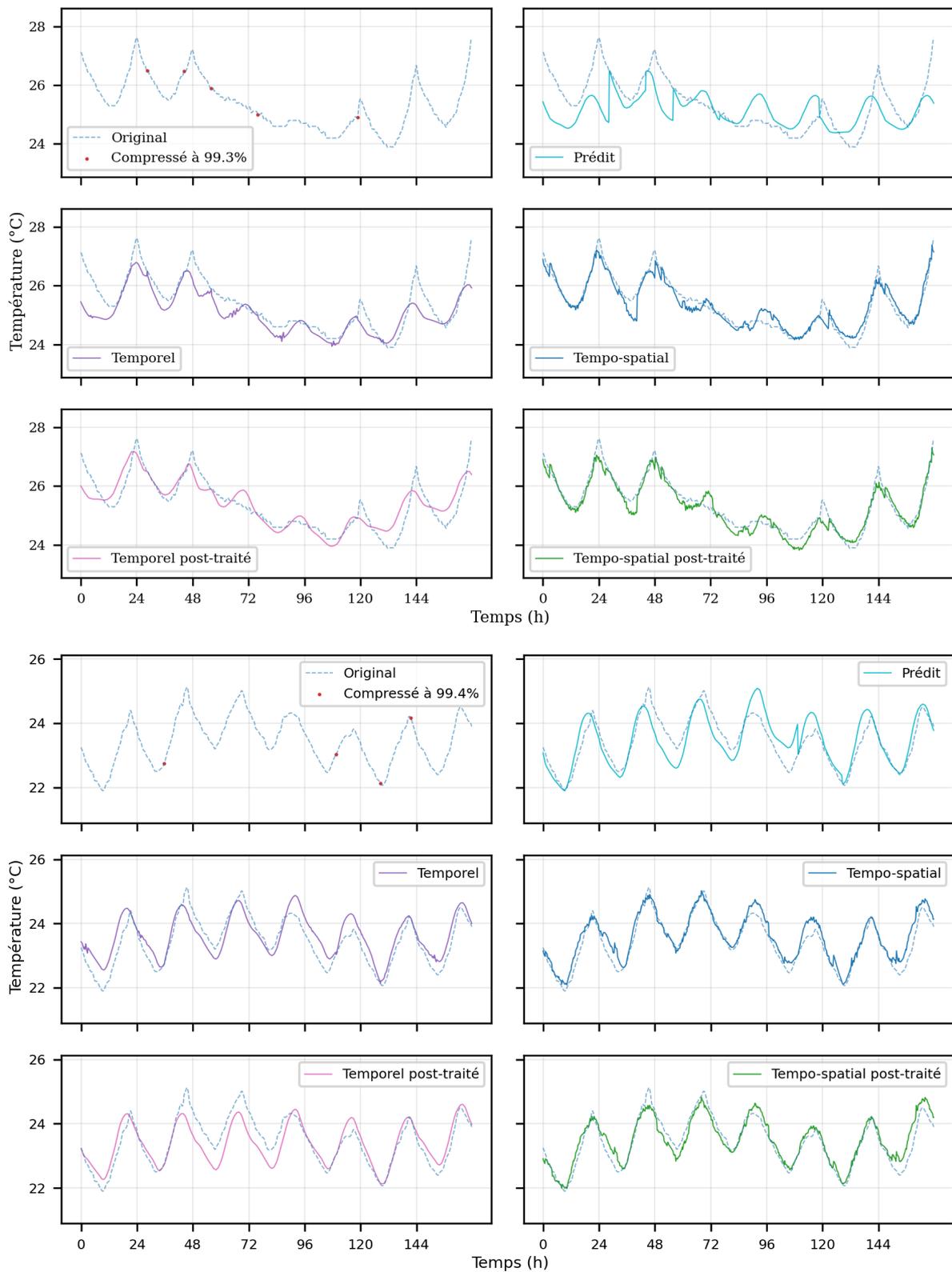


Figure II.35: Exemple de reconstructions de signaux tests comprimés à 99,3% et 99,4%

Chapitre II – Développement de méthodes de reconstruction des signaux comprimés par apprentissage profond – application au réseau de capteurs

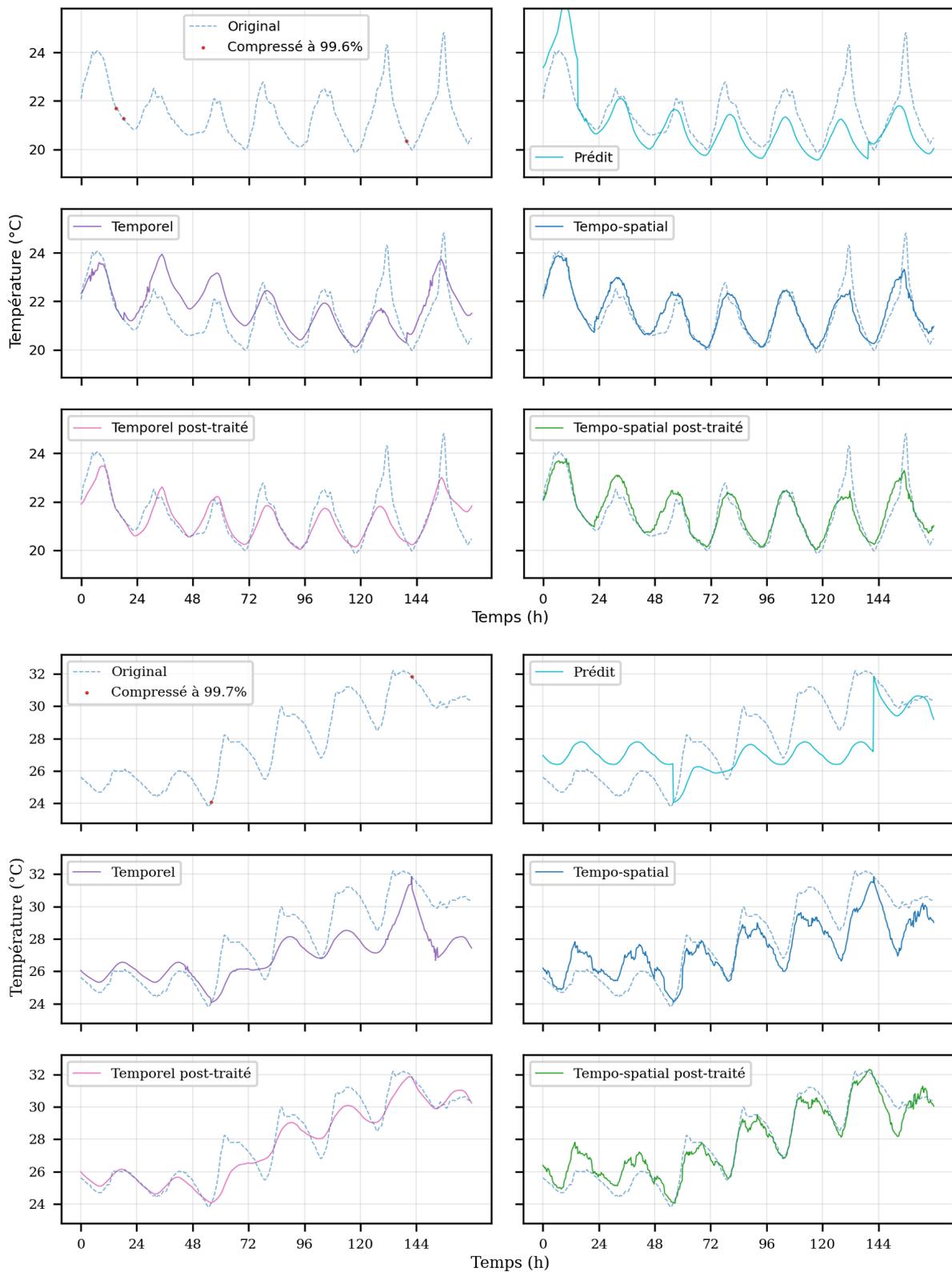


Figure II.36: Exemple de reconstructions de signaux tests comprimés à 99,6% et 99,7%

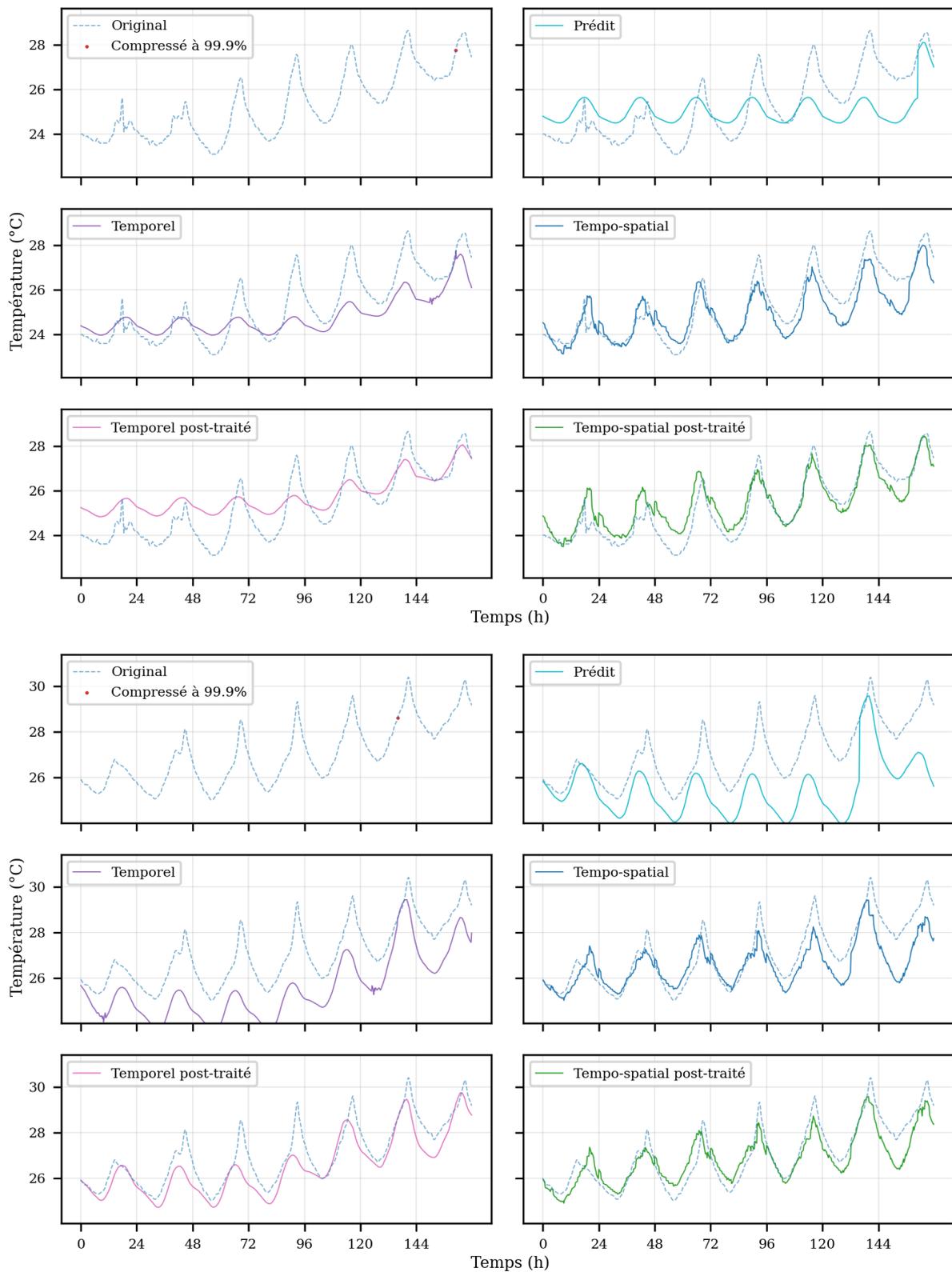


Figure II.37: Exemple de reconstructions de signaux tests comprimés à 99,9%

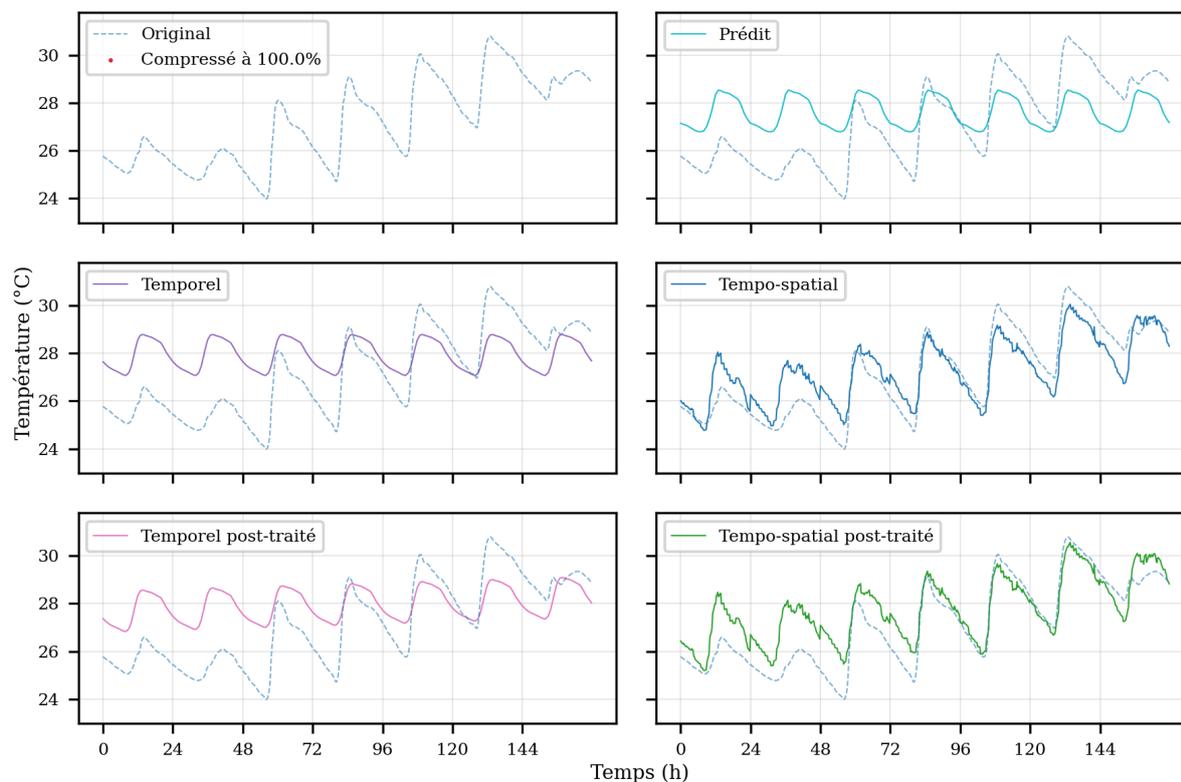


Figure II.38: Exemple de reconstruction d'un signal test compressé à 100%

Le cas d'un signal compressé à 100% comme dans la Figure II.38 peut survenir lorsque le taux global de compression de l'ensemble des signaux est très élevé. Dans cet exemple, l'ensemble des signaux est compressé à 99.7% et le signal compressé au taux le plus bas l'est à 99.1%. Alors la reconstruction d'un signal compressé à 100% s'appuie essentiellement sur les corrélations spatiales et les valeurs réelles des autres signaux de l'ensemble. Dans l'exemple, on observe bien que la reconstruction temporelle seule ne parvient pas à ressembler au signal d'origine, ce qui n'est pas le cas lorsque au moins une valeur réelle existe dans le signal compressé.

II.3 Bilan

Les méthodes temporelle et tempo-spatiale présentées surpassent les méthodes de l'état de l'art d'AC appliquées aux RCSFs selon des critères de précision et de temps de reconstruction. Si nos méthodes se distinguent par leur efficacité et leurs performances c'est principalement en raison de leurs structures qui sont conçues pour remplir un objectif clair: saisir et enregistrer les corrélations, tant temporelles que spatiales, contenues dans un jeu de données raster. Elles mettent en coopération et en relation directe des RNRs et RNCs, notamment grâce à la caractéristique de fiabilité des prédictions, en accomplissant un challenge que pose la littérature en 2023 (cf I.2.2.3), à savoir la combinaison de ces derniers RNs.

Les résultats obtenus par la méthode temporelle ont permis l'émancipation vis-à-vis de l'AC dans ce contexte d'application de signaux qu'on qualifie de pseudo-aléatoires. Cependant, si une représentation parcimonieuse de quelques paramètres est possible, dans le cas de signaux périodiques plus constants ou de problèmes linéaires, on préfère de loin employer des méthodes AC plus efficaces et qui seront également plus précises que des méthodes d'AP.

Les modèles de RNs employés dans les différentes méthodes de reconstructions peuvent aussi être utilisés individuellement (à l'exception évidente du discriminateur) pour réaliser une reconstruction spécifique. Les méthodes peuvent être implémentées ou utilisées partiellement en fonction de critères comme la rapidité de traitement, la mémoire vive disponible, le taux de compression des signaux ou la marge d'erreur de reconstruction tolérée. Ces modèles ont été pensés pour être généralisés aux données raster et aux STs sans se limiter aux signaux de RCSFs. L'IPA deepsignals a été conçue et développée dans ce sens pour permettre des applications diverses. Une telle généralisation est possible si les données peuvent être implicitement représentées de façon parcimonieuse. Pour faire un parallèle avec la théorie de l'AC, on peut dire que les RNs apprennent simultanément les bases de représentation et les modèles de reconstruction des données sous forme de boîte noire. Ce qui veut dire que nous ne pouvons pas extraire et formuler formellement ces modèles (le concept de boîte noire est repris et détaillé dans le chapitre suivant III).

Les applications ont démontré que les méthodes à base de RAG ne sont pas bien adaptées au problème des signaux compressés malgré des temps de calculs plus courts. Comparativement aux autres modèles implémentés, les reconstructions développées

au cours de ces travaux sont stables dans la qualité de leurs résultats à de hauts taux de compression tandis que l'erreur augmente exponentiellement chez les autres méthodes. En connectant un RCSF à un serveur qui exécute les techniques d'AP de la thèse, l'acquisition des mesures auprès des noeuds de capteurs peut être commandée de façon à préserver l'intégrité énergétique et économique du système.

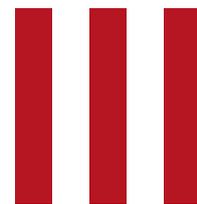
La limite des RNRs entraînés pour une application à un RCSF est qu'ils ne connaissent que la nature des signaux qu'ils ont vus à l'entraînement. Si les mesures évoluent en conséquence de modifications de l'environnement du RCSF tel que le réchauffement climatique, des travaux, du nouveau matériel, par exemple, les RNs nécessiteront un nouvel entraînement comme une mise à jour. C'est une problématique à laquelle cette thèse peut répondre sans nécessité d'intervention humaine. Il se trouve que le discriminateur parvient à corrélérer l'erreur de reconstruction avec la vraisemblance des signaux. Mieux que cela, la fonction du discriminateur est, avant toute chose, d'évaluer le niveau de proximité entre des signaux traités par les modèles de RNs et des signaux connus à l'entraînement comme étant réels. Il offre la possibilité d'automatiser la commande de mise à jour des RNs si jamais il ne détecte plus de vraisemblance (à partir d'un certain seuil, pendant un certain temps) chez les nouveaux signaux reconstruits. C'est un pilotage automatique d'acquisition des signaux et d'apprentissage actif des RNs qui reste à développer mais qui permettrait la pérennité du système de reconstruction des signaux compressés.

A la suite de ces travaux, plusieurs perspectives sont possibles:

- Afin d'appuyer les contributions à l'état-de-l'art de l'AP, il est judicieux d'implémenter le modèle PriSTI et le modèle PredRNN et sa cellule ST-LSTM et de les adapter pour accomplir la tâche de reconstruction de signaux compressés de RCSF afin de comparer leurs résultats avec ceux de la méthode tempo-spatiale. Cette étude sera menée à l'issue de la rédaction de cette thèse en vue d'une prochaine publication.
- L'idée précurseuse introduite dans la première sous-section II.1.1.1 peut, sur le plan théorique, apporter une contribution majeure à l'état-de-l'art d'AC. Des travaux basés sur l'exploitation de cette idée nous semble être une perspective prometteuse.
- Les méthodes présentées traitent uniquement des signaux déjà compressés. Elle accomplissent une reconstruction à partir d'acquisitions aléatoires de valeurs

réelles. Dans la logique de l'AC, la possibilité d'une co-conception d'un système intelligent d'acquisition et de reconstruction des signaux d'un RCSF semble être la prochaine piste pour améliorer la précision en fonction du taux de compression.

- Il sera intéressant de dresser un bilan énergétique et économique au niveau de la consommation et l'autonomie des noeuds de capteurs lors de la mise en oeuvre des méthodes présentées pour observer les gains que peuvent offrir ces travaux de thèse. En théorie, du point de vue de l'autonomie des capteurs, le gain énergétique est quasiment proportionnel au facteur de réduction du nombre d'échantillons nécessaires pour une reconstruction fidèle du phénomène à mesurer.



Contribution à l'identification de modèles de simulations de structures colloïdales

Sommaire

III.1	Les simulations de structures colloïdales	115
III.1.1	Les activités de l'IRCER	115
III.1.1.1	Présentation de l'IRCER	115
III.1.1.2	Colloïdes	117
III.1.1.3	Les problématiques de la thèse	119
III.1.2	Le potentiel d'interaction de particules Janus en suspension .	121
III.1.2.1	Les formules de calculs	121
III.1.2.2	Les paramètres de simulations	123
III.1.2.3	Les particules framboise	125
III.1.3	L'identification paramétrique	126
III.1.3.1	Principe et modèles	126
III.1.3.2	Boîte grise	127
III.2	Application	129
III.2.1	Développement des modèles	129

III.2.1.1	Les données de simulation de l'IRCER	129
III.2.1.2	Le modèle d'apprentissage profond de simulateur .	130
III.2.1.3	Le modèle de générateur de paramètres	131
III.2.2	Tests	134
III.2.2.1	Preuve de concept	134
III.2.2.2	Cas d'étude de l'IRCER	135
III.2.2.3	Analyse des résultats	138
III.3	Bilan	144

Ce chapitre traite entièrement des travaux menés en collaboration avec l'IRCER de Limoges au sein du projet du LabEx Σ -LIM. Nous avons travaillé avec Arnaud Videcoq, professeur des Universités et responsable de l'axe « Procédés céramiques » de l'IRCER. La littérature et les objectifs de ces travaux sont soumis à son expertise.

III.1 Les simulations de structures colloïdales

III.1.1 Les activités de l'IRCER

III.1.1.1 Présentation de l'IRCER

L'IRCER est le plus grand laboratoire français de céramique. Il est basé à Limoges, ville connue pour son industrie de la céramique, au Centre Européen de la Céramique. Cet institut est un acteur important et novateur dans le contexte économique et écologique actuel. L'exploitation du matériau céramique remonte à la préhistoire, ce terme désigne aujourd'hui une pléthore d'objets composés de matériaux minéraux. Le procédé céramique, au sens de la technique, est l'art de la fabrication de ces objets. On y retrouve notamment la fabrication de poterie, faïence, porcelaine, grès, etc. L'aspect fonctionnel de ce matériau s'étend à de nombreux domaines industriels comme le bâtiment, les prothèses médicales, l'équipement des usines (fours, cheminées, outils), l'équipement de maison (sanitaire, vaisselle, décoration). L'IRCER étudie et maîtrise les propriétés et techniques de la céramique depuis l'atome jusqu'à l'objet pour synthétiser de nouveaux matériaux.

L'approche et le développement de l'IRCER qui mêlent acteurs pluridisciplinaires, partenariats entre laboratoires et procédés innovants permet d'introduire la céramique dans les nouvelles technologies pour des applications dans les domaines des éco-matériaux, de l'électronique, de l'optique, des télécommunications, de l'aviation et l'aérospatial, de l'énergie et du bio-médical. Ses axes de recherches sont les procédés céramiques, les procédés plasmas et lasers, l'organisation structurale multi-échelle des matériaux et les céramiques sous contraintes environnementales. Voici une liste non exhaustive de contributions à la recherche et l'industrie que l'on doit à l'IRCER :

- Miniaturisation des composants : élaboration par ablation laser, PVD (Physical Vapor Deposition) ou PECVD (Plasma Enhanced Chemical Vapor Deposition) de couches minces pour composants et systèmes micro-électro-mécaniques.

- Conception et développement de procédés d'élaboration de matériaux nanocomposites multifonctionnels, notamment la stéréolithographie et l'impression jet d'encre.
- Nouvelles architectures de composants : élaboration par fabrication additive (stéréolithographie, impression jet d'encre, micro-extrusion 3D. . .) de composants diélectriques micro-ondes, de circuits à trois dimensions et de capteurs multifonctionnels.
- Nouveaux matériaux vitreux à base d'oxyde de tellure présentant une non-linéarité optique des deuxième et troisième ordres et élaboration de fibres et/ou de guides d'ondes à partir de ces matériaux.
- Matériaux céramiques polycristallins transparents de grande dimension pour amplification laser de puissance.
- Nouveaux composés oxydes piézoélectriques / ferroélectriques sans plomb Conception de matériaux céramiques sans substance toxique (e.g. en accord avec la norme REACH).
- Substitution des additifs issus de la pétrochimie par des additifs biosourcés.
- Développement à partir de céramiques, de nouveaux matériaux durables, recyclables, dont les déchets sont valorisables.
- Mise au point de procédés de fabrication à base d'eau et sans cuisson afin de minimiser l'impact environnemental.
- Matériaux hautes températures en environnements agressifs (e.g. nouvelles générations de réacteurs nucléaires, chambres de combustion de réacteurs d'avion ou d'engins spatiaux, vaporeformage du méthane).
- Nouvelles générations de barrières thermiques (e.g. aubes de turbine de réacteur d'avion).
- Revêtements multimatériaux pour applications tribologiques.
- Systèmes catalytiques pour la production de gaz de synthèse ou d'hydrogène à partir du gaz naturel (Laboratoire Commun Air Liquide).
- Nouveaux matériaux d'électrolyte pour SOFC (700 °C). Élaboration de cœurs de

pires et d'EHT (Electrolyse Haute Température).

- Nouveaux matériaux pour la récupération (e.g. photovoltaïque) et le stockage d'énergie (e.g. systèmes hybrides entre batteries et supercondensateurs) Synthèse de différents phosphates de calcium. Surfaces adaptées au greffage.
- Fonctionnalisation des biocéramiques par des molécules actives pour stimuler les processus de régénération osseuse (greffage de peptides) et/ou pour un traitement thérapeutique (inclusion et libération d'antibiotiques, antiviraux, antitumoraux...).
- Élaboration d'implants à microstructure et architecture spécifiques pour des applications en ingénierie tissulaire osseuse
- Biocapteurs pour le diagnostic et le traitement des tumeurs précoces.

Pour plus d'informations et de détails concernant les activités de l'IRCER, voici le lien vers leur site web : <https://www.ircer.fr/>.

III.1.1.2 Colloïdes

Un colloïde est une particule de petite taille (de quelques nanomètres à quelques micromètres) qui est maintenue en suspension au sein d'un liquide suspensif. Une suspension colloïdale est le nom donné à une dispersion de colloïdes comme illustré Figure III.1.

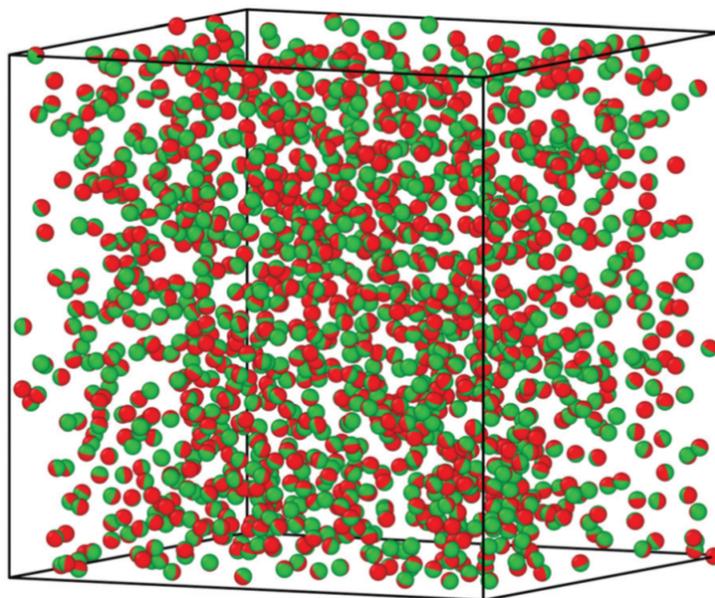


Figure III.1: Modélisation de nanoparticules Janus (particules à deux faces) en suspension dans un milieu de dispersion (figure issue de [101])

Ce mélange est hétérogène et l'amas de particules est visible à l'œil nu contrairement aux solutions qui sont des mélanges homogènes avec des particules de taille inférieure au nanomètre. L'agitation thermique maintient la suspension colloïdale, elle est prépondérante par rapport à la gravité empêchant la précipitation des particules. Les interactions attractives et répulsives qu'exercent les particules entre elles conditionnent la stabilité de la suspension. On parle de stabilité lorsque que les forces en vigueur résultent en un équilibre qui empêche la sédimentation.

La suspension colloïdale entre dans les procédés de fabrication de matériau céramique. Le principe de base consiste à mélanger la poudre solide avec un solvant liquide pour former une suspension ou une pâte qui est ensuite mise en forme. La fabrication par voie colloïdale de l'IRCER suit les étapes suivantes :

1. Synthèse de la poudre
2. Préparation de la suspension
3. Mise en forme : fabrication de l'objet céramique par le procédé choisi (coulage, stéréolithographie, impression jet d'encre, etc)
4. Densification (à haute température)

Dans ce procédé, la suspension est suffisamment instable pour que les particules s'agrègent comme illustré Figure III.2. La stabilité des suspensions est recherchée la plupart du temps dans les procédés mais l'agrégation est étudiée aujourd'hui pour exploiter l'auto-assemblage de la matière au cours de la mise en forme.

La poudre est synthétisée de telle sorte que les colloïdes soient, relativement aux molécules du solvant, assez grandes pour être soumises au mouvement brownien et assez petites pour être maintenues en suspension. Le mouvement brownien est un mouvement aléatoire chaotique qui est observé sur une particule immergée dans un fluide. Ce mouvement résulte des collisions de la particule immergée avec les particules plus petites qui composent le fluide, on ne sait le reproduire que d'un point de vue statistique.

Dans la suite du chapitre, on ne considère que le genre de suspension colloïdale pratiquée par l'IRCER pour la céramique.

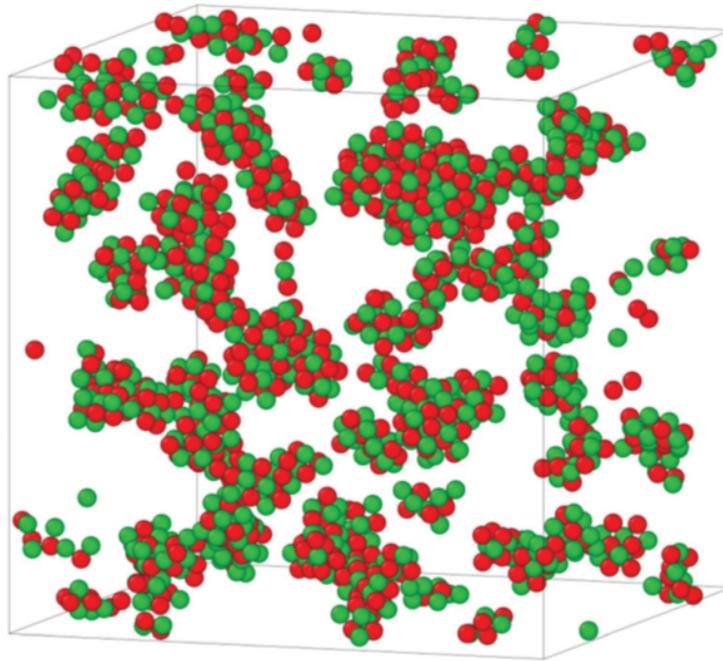


Figure III.2: Modélisation d'agrégats de particules dans une suspension colloïdale (figure issue de [101])

III.1.1.3 Les problématiques de la thèse

Le projet scientifique du LabEx Σ -LIM dans lequel s'investissent l'IRCER et XLIM porte l'ambition de développer de nouveaux procédés céramiques pour la fabrication de composants électroniques et photoniques dédiés à des applications dans les télécommunications. L'objectif est de produire des avancées technologiques tout en améliorant le niveau d'autonomie des systèmes du point de vue énergétique. Dans ce contexte, l'IRCER souhaite apporter des modifications à son logiciel de simulation pour contrôler avec précision l'agrégation dans les structures colloïdales des matériaux céramiques. La simulation des mouvements et interactions des particules permet l'identification des conditions initiales de fabrication par voie colloïdale qui conduisent au matériau escompté, avec des propriétés mécaniques, thermiques et électriques spécifiques. Les simulations de suspension colloïdales de l'IRCER en 2023 sont extrêmement coûteuses en mémoire et souffrent de temps de calculs particulièrement longs malgré une optimisation des calculs sur le plan algorithmique et mathématique. L'IRCER est à la recherche d'améliorations ou de nouveaux modèles de simulations pour corriger ces imperfections.

Les calculs des simulations sont basés sur la théorie de Langevin qui implique des forces d'interactions entre particules, des forces de friction avec le solvant et des forces aléatoires de mouvement brownien. L'IRCER, à l'initiative de Monsieur Arnaud Videcoq,

professeur des Universités et responsable de l'axe « Procédés céramiques » de l'IRCER, a collaboré aux travaux de thèse pour exploiter les outils d'AP dans l'élaboration d'un modèle de simulation de structures colloïdales. Son idée est d'exploiter les formules de potentiel d'interactions des particules Janus de la littérature en identifiant les paramètres optimaux à la description de systèmes colloïdaux par le biais de RNs. Ce travail vise à réduire la complexité des modèles de simulations tout en conservant un niveau de précision au moins équivalent à celui des résultats de simulations d'origine. Les travaux de thèse concernent la partie conception et développement de modèles d'AP tandis qu'Arnaud Videcoq a fourni les données de simulations de mouvements de particules, les formules mathématiques et son expertise de la littérature et des résultats.

III.1.2 Le potentiel d'interaction de particules Janus en suspension

Les particules anisotropes dites « Janus » sont des micro ou nanoparticules à deux faces, comme illustré sur la Figure III.3. Chacune de ces faces possède ses propres propriétés physiques et chimiques de surface. Elles sont appelées ainsi en référence à Janus, le dieu romain aux deux visages. Dans le cadre de la thèse, les particules Janus possèdent un hémisphère électriquement chargé positivement et l'autre négativement. Le vecteur unitaire \vec{u} donne l'orientation de la particule, plus spécifiquement de son hémisphère positif.

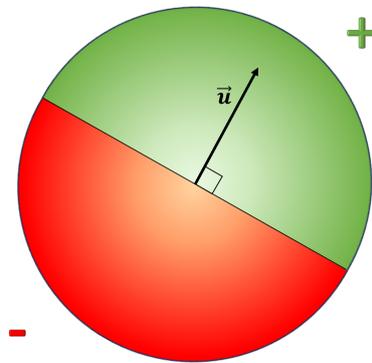


Figure III.3: Modélisation d'une particule Janus

III.1.2.1 Les formules de calculs

L'objet de cette section n'est pas d'expliquer la physique des particules, que ce soit dans ses fondamentaux ou dans le cas de particules Janus en suspension. Le sujet de ces travaux est l'identification paramétrique de modèles de simulations de structures colloïdales. L'expertise mathématique et physique revient à l'IRCER, en la personne d'Arnaud Videcoq. A noter en particulier qu'il en est de même concernant la critique de l'état de l'art et le choix des formules d'interactions utilisées. Ce manuscrit se positionne sur la partie technique de l'AP. C'est pourquoi cette section se limite à présenter les informations nécessaires à la compréhension de ce travail, de ces objectifs et des données d'apprentissage.

Les collisions des particules Janus avec celles du solvant, plus petites, provoquent un mouvement brownien, donc aléatoire. Cette dynamique apporte un caractère imprévisible à la formation d'agrégats dans une suspension colloïdale. Le comportement

aléatoire du mouvement des particules, qui est bien réel, implique que deux réalisations indépendantes, qu'elles soient expérimentales ou numériques, ne peuvent pas produire exactement les mêmes agrégats, mais leurs propriétés d'ensemble ou statistiques seraient bien les mêmes. La théorie de Langevin du mouvement brownien est qu'une particule immergée dans un fluide à un instant t est soumise à deux forces: un frottement dû à la viscosité du fluide, noté f , et une force aléatoire résultant des chocs avec les molécules du fluide suffisamment forte pour conserver la particule en mouvement malgré le frottement, notée η . Selon le principe fondamental de la dynamique de Newton, voici l'équation stochastique de Langevin:

$$m \frac{dv(t)}{dt} = f(t) + \eta(t) \quad (\text{III.1})$$

Où m est la masse de la particule et v sa vitesse dans le fluide. Or, si l'on considère des particules en interaction comme sur la Figure III.4, elles sont dites couplées, elles exercent l'une sur l'autre une force et un couple qui s'ajoutent à cette dynamique de Langevin et vient agir sur leur vitesse de déplacement et de rotation. Le mouvement des particules en suspension est donc soumis, d'une part, au mouvement brownien et, d'autre part, aux interactions avec leurs pairs.

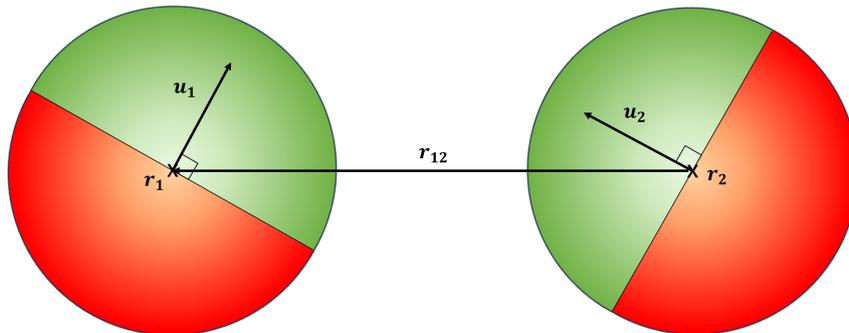


Figure III.4: Modélisation de deux particules Janus en interaction

r_{12} est le vecteur entre les deux centres de particules en interaction.

Dans la littérature de physique des particules, plusieurs travaux sont consacrés à l'établissement de lois mathématiques sur les interactions et mouvement des particules Janus en suspension. Malheureusement, le caractère anisotrope des particules et le mouvement brownien compliquent rudement cette tâche et aucun modèle n'est

généralisable à ce jour. L'équipe de l'IRCER s'intéresse à démontrer qu'une formule de la littérature peut fidèlement simuler les interactions et mouvement des particules Janus en suspension si elle est correctement paramétrée. On s'intéresse à la formule III.2 de Rosenthal [102] qui définit l'interaction ϕ^I entre les hémisphères des particules Janus et à sa variante mise au point à l'IRCER, l'équation III.3.

$$\phi^I(\mathbf{r}_{12}, \mathbf{u}_1, \mathbf{u}_2) = C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} (\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{r}_{21} \quad (\text{III.2})$$

$$\begin{aligned} \phi^I(\mathbf{r}_{12}, \mathbf{u}_1, \mathbf{u}_2) = C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \\ \left(a_1 \left(\mathbf{u}_1 \cdot \frac{\mathbf{r}_{21}}{|\mathbf{r}_{21}|} * \mathbf{u}_2 \cdot \frac{\mathbf{r}_{21}}{|\mathbf{r}_{21}|} \right) + \left(a_2 - \frac{a_1}{3} \right) \mathbf{u}_1 \cdot \mathbf{u}_2 \right) \end{aligned} \quad (\text{III.3})$$

Le potentiel d'interaction ϕ entre deux particules que l'on considère dans ce cas s'exprime alors comme dans l'équation III.4 suivante.

$$\phi(\mathbf{r}_{12}, \mathbf{u}_1, \mathbf{u}_2) = \phi^{soft}(|\mathbf{r}_{12}|) + \phi^I(\mathbf{r}_{12}, \mathbf{u}_1, \mathbf{u}_2) \quad (\text{III.4})$$

Où σ est le diamètre des particules et ϕ^{soft} traduit la force de répulsion à courte distance qui permet aux particules qui entrent en collision de conserver leur intégrité de forme, tel que:

$$\phi^{soft}(|\mathbf{r}_{12}|) = 4\epsilon \left(\frac{\sigma}{|\mathbf{r}_{12}|} \right)^{12} \quad (\text{III.5})$$

Les autres paramètres sont expliqués dans la section suivante, ils n'ont pas de valeurs constantes connues à ce jour. C'est le problème d'optimisation paramétrique sous contrainte que ce travail cherche à résoudre.

III.1.2.2 Les paramètres de simulations

Selon la littérature, les paramètres variables de simulations sont dépendants des conditions dans lesquels ils entrent en jeu. La température et le diamètre des particules en suspension sont des valeurs indicatives importantes qui influent sur le potentiel

d'interaction des particules anisotropes.

En s'appuyant sur les travaux de Rosenthal et de l'IRCER, les valeurs des paramètres qui doivent être précisées sont celles de C , ϵ et λ :

Le paramètre C mesure la force de couplage, attractive ou répulsive, il caractérise l'intensité de l'interaction. Les travaux de Rosenthal estiment que sa valeur devrait être équivalente au rapport III.6.

$$C = \frac{k_B T \sigma}{T_{agg}} \quad (\text{III.6})$$

Où T_{agg} est la température spécifique aux particules en dessous de laquelle une agrégation peut se produire. k_B est la constante de Boltzmann qui vaut $1,380649e^{-23} J \cdot K^{-1}$, elle fait le lien entre la température thermodynamique et l'énergie à l'échelle microscopique. Pour sa part, l'IRCER estime que la force de couplage C des particules en suspension de céramique doit se trouver entre les valeurs de $3k_B T \sigma$ et $15k_B T \sigma$.

Le paramètre ϵ est une constante de couplage d'intensité de répulsion à courte distance qui empêche le chevauchement théorique des particules. Elle est évaluée à ce jour selon le rapport de $\frac{k_B T}{\epsilon} = 1$. L'IRCER estime que sa valeur pour les suspensions colloïdales de céramique doit se trouver entre $2k_B T$ et $10k_B T$.

Le paramètre λ caractérise la portée de l'interaction. Plus la valeur de λ est petite, plus la portée d'interaction est grande et plus la taille des agrégats augmente. Le modèle de Rosenthal est conçu dans le cadre de la théorie classique de la densité fonctionnelle qui estime la valeur de λ comprise entre $\frac{2}{\sigma}$ et $\frac{4}{\sigma}$. Quant à l'IRCER et ses simulations ces bornes seraient $\frac{1}{\sigma}$ et $\frac{5}{\sigma}$. A l'extérieur de ces intervalles de valeurs, les simulations donnent lieu à des résultats physiquement invraisemblables.

a_1 et a_2 sont des paramètres de pondération uniques à la formule de l'IRCER. a_1 évalue le caractère amphotère des interactions et, à ce titre, doit être négatif. Le paramètre a_2 est ajusté en fonction de la valeur de a_1 . L'IRCER a mené une étude sur les valeurs de ces paramètres et a conclut que les valeurs comprises entre les bornes suivantes permettaient des agrégations physiquement et expérimentalement convaincantes et que leur présence au sein d'une formule dérivée du modèle de Rosenthal améliorerait sensiblement les résultats de simulations: $-4 < a_1 < -1$ et $-2 < a_2 < 3$.

III.1.2.3 Les particules framboise

L'IRCER et l'équipe d'Arnaud Videcoq ont mis au point un nouveau modèle, publié en 2019 [101], de simulation de dynamique brownienne à gros grains. Il permet d'expliquer la diversité des formations d'agrégats dans une suspension en discrétisant dans une inhomogénéité les propriétés de surface d'une particule. Chaque particule est modélisée comme une géode de 92 sphères ce qui leur donne l'apparence et la nomination de particule « framboise », voir Figure III.5.

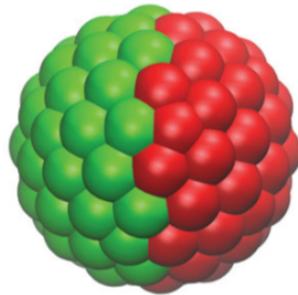


Figure III.5: Modélisation d'une particule framboise (figure issue de [101])

Contrairement aux particules Janus à deux hémisphères homogènes pour lesquelles les modèles théoriques actuels ne parviennent pas à simuler les mouvements qui s'observent dans les expérimentations, la modélisation en particules framboises apporte de meilleurs résultats comparatifs qui sont étudiés dans ce papier [103]. Cette représentation exploite le caractère anisotrope des particules en calculant les interactions locales de leur surface. C'est le modèle de simulation actuel de l'IRCER. Son défaut majeur est sa gourmandise qui requiert autant de calculs qu'une particule possède de sphères élémentaires. Chaque sphère a son propre potentiel électrique de surface et interagit avec toutes les autres, sauf avec celles de sa propre particule. C'est pourquoi l'intérêt principal des travaux de thèse est de reproduire la qualité des résultats de simulation atteinte avec une modélisation framboise des particules par le biais d'une modélisation plus simple et moins coûteuse (cf formule III.4 : des particules Janus avec deux hémisphères homogènes).

Les références citées [101], [103] livrent beaucoup plus de détails ainsi que les théories physiques et modèles mathématiques utilisés dans cette modélisation framboise.

III.1.3 L'identification paramétrique

III.1.3.1 Principe et modèles

L'identification Paramétrique (IP) ou identification de système est un terme générique qui rassemble les méthodes automatiques qui permettent de déduire un modèle mathématique à partir de mesures expérimentales d'un système, qu'il soit linéaire ou non linéaire. En général, la structure du modèle mathématique est connue et l'objectif est d'identifier les valeurs de paramètres de transfert du modèle, d'où son appellation. Le processus d'identification est très similaire au processus d'apprentissage des RNs. Il est basé sur une boucle qui applique une méthode d'identification d'un modèle mathématique à des mesures expérimentales jusqu'à ce qu'une fonction de validation, qui considère des mesures différentes, approuve le modèle. L'analogie est illustrée Figure III.6.

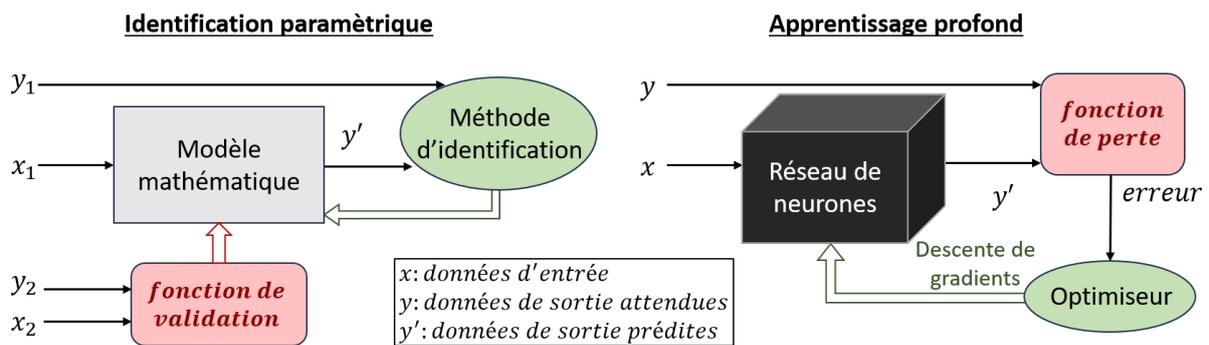


Figure III.6: Similitudes entre identification et apprentissage

La représentation en couches hiérarchiques des données au sein d'un RN lui permet d'apprendre de tout type de données pour résoudre tout type de problème mais c'est aussi ce qui le rend trop complexe et imprécis pour modéliser des systèmes linéaires simples. Toujours est-il que l'AP est capable d'IP mais ne s'y limite pas comme le met en évidence le tableau III.1.

La fonction de transfert est une notion surtout employée dans le traitement du signal pour décrire la relation entre les entrées et les sorties d'un système. Elle permet de caractériser la réponse impulsionnelle. Dans le cadre de l'IP et plus largement dans les domaines mathématique et automatique, la fonction de transfert désigne une transformée en Z (une transformation de Laplace pour un ensemble fini de valeurs) qui permet de modéliser la relation entre des conditions initiales et des mesures expérimentales dans un domaine observable et discret. Les modèles les plus connus du genre pour l'IP sont les modèles autorégressifs de type Auto Regressive model with

Domaine	IP	AP
Automatisation	Modérée	Élevée
Problèmes	Régression	Tous
Type de données	Mesures	Tous
Quantité de données	Modérée	Élevée
Modèle	Fonction de transfert (souvent)	Couches hiérarchiques
Optimisation	Méthode des moindres carrés (souvent)	Descente de gradients stochastiques

Table III.1: Comparatif des caractéristiques d'IP et d'AP

eXternal inputs (ARX) qui ajoutent un bruit blanc, donc gaussien à moyenne nulle, aux données d'entrées pour modéliser des fluctuations non mesurables du système.

III.1.3.2 Boîte grise

Le terme de boîte dans la littérature est né du besoin d'introduire le concept de boîte noire pour le dissocier de ce que l'on appelle par opposition une boîte blanche. Une boîte désigne un processus, un modèle, qui traite des informations. Une boîte noire est non interprétable par l'Homme mais conçu par lui dans un but précis. La métaphore d'une boîte noire représente l'idée que l'Homme, en observant le fonctionnement de cette dernière, ne peut pas voir, comprendre, de quelle façon elle parvient à réaliser sa tâche. L'intérêt d'un tel modèle est de produire un résultat attendu, non d'exprimer formellement comment ce résultat est produit. D'une manière générale, les RNs sont des boîtes noires. Les hyperparamètres permettent de les structurer et l'apprentissage de les émanciper de l'intervention humaine. Ce sont de puissants modèles de l'AA que l'Homme ne peut pas traduire ou formuler soi-même. De l'autre côté, les boîtes blanches désignent ce que l'on sait exprimer et manipuler, comme par exemple des lois de la physique sous forme d'équations mathématiques qui ont fait l'objet de démonstrations.

Ces boîtes sont complémentaires dans la littérature scientifique et dans l'industrie, les blanches permettent des avancées dans la compréhension et la maîtrise des lois de l'univers tandis que les noires permettent des avancées pour l'automatisation et

l'autonomie des systèmes. Les boîtes noires sont apparues pour accomplir ce que les chercheurs n'arrivent pas encore à exprimer avec des boîtes blanches. Aujourd'hui, elles ont prouvé leur efficacité dans de nombreux domaines et applications. L'interprétation de ces boîtes permettrait des avancées majeures dans les domaines qui les utilisent et notamment dans la modélisation mathématique de phénomènes physiques ou sociaux. Dans cette optique, la littérature a introduit un nouveau concept de boîte grise. Ce terme désigne toutes les méthodes visant à interpréter ou à rendre observable des éléments de boîte noire. Une boîte grise est une boîte blanche qui contient des sous-éléments en boîte noire ou vice et versa. Cela permet, entre autres, de mêler connaissances fondamentales et automatisme, de tester des modèles expérimentaux, de réaliser de l'IP. On reconnaît cinq catégories de boîte grise :

- La boîte noire contrainte: approche dans laquelle des connaissances sont incorporées sous la forme de restrictions supplémentaires imposées à la boîte noire d'origine.
- La modélisation semi-physique: les données d'entrée et de sortie du système peuvent subir une transformation basée sur un modèle connu à l'intérieur d'une boîte noire.
- La modélisation mécaniste: un modèle de boîte blanche enrichi par une évaluation en boîte noire des erreurs du modèle.
- La modélisation hybride: un modèle explicitement divisé en parties de boîtes noires et boîtes blanches.
- Le système de paramètres distribués: une approche visant à distinguer les paramètres valides au sein d'un modèle non valide.

Cette thèse [104] étudie plus largement l'ensemble de ces concepts de boîtes et leur implication dans l'IP avec AP.

Dans le cadre des travaux de thèse en collaboration avec l'IRCER, l'objectif est d'optimiser des modèles de simulations qui entrent dans la catégorie des boîtes blanches via des modèles d'AP qui entrent dans la catégorie des boîtes noires. Nos applications explorent diverses solutions de boîtes grises essentiellement basées sur de la modélisation hybride, mais incluant aussi une boîte noire contrainte et une modélisation semi-physique.

III.2 Application

III.2.1 Développement des modèles

III.2.1.1 Les données de simulation de l'IRCER

Un jeu de données issu des simulations de structures colloïdales fourni par l'IRCER est constitué de 20000 fichiers qui contiennent, pour un instant t , les vecteurs de position, d'orientation, de force et de couple de 1500 particules Janus. Pour rappel, les particules sont modélisées en framboises pendant les simulations (cf III.1.2.3). Leur diamètre est de 600nm. Le tableau III.2 est un exemple de valeurs issu d'un fichier pour une particule.

Axe	Position (m)	Orientation	Force (N)	Couple (N.m)
x	$6.59382e^{-6}$	$-5.39663e^{-1}$	$-1.35664e^{-14}$	$-9.27919e^{-23}$
y	$1.03531e^{-5}$	$-1.33621e^{-1}$	$-2.00623e^{-14}$	$-2.72045e^{-22}$
z	$5.106e^{-6}$	$8.3121e^{-1}$	$-6.19007e^{-14}$	$1.20599e^{-22}$

Table III.2: Échantillon de données pour une particule

Au commencement d'une simulation, les particules sont initialisées à des positions et orientations aléatoires à l'échelle microscopique dans un espace confiné qui représente le contenant dans lequel se trouve le solvant. La simulation calcule un vecteur de force et de couple pour chaque particule qui sont les sommes respectives des forces et des couples qu'exercent toutes les autres particules sur chacune d'entre elles. On parle de faire la somme des interactions entre les paires de particules. Ces vecteurs déterminent les prochains mouvements des particules. Il y a 1000 fichiers par temps de simulation différents, qui sont: 0.01s, 0.1s, 0.2s, 0.3s, 0.4s, 0.5s, 0.6s, 0.7s, 0.8s, 0.9s, 1s, 2s, 3s, 4s, 5s, 6s, 7s, 8s, 9s, 10s. Ces temps servent à entraîner le RN à partir de différents cas de mouvements et d'agrégations des particules. Dans un premier temps, les particules interagissent peu entre elles car elles sont dispersées. Dans un second temps, chaque particule possède et interagit avec plusieurs particules voisines qui se sont agrégées ensemble. Les 1000 fichiers de données fournis à chacun des temps correspondent à l'évolution de 1000 configurations initiales différentes pour observer une grande variété de formation d'agrégats.

III.2.1.2 Le modèle d'apprentissage profond de simulateur

Les données disponibles n'incluent pas les valeurs des paramètres à identifier. Or, un RN requiert des données cibles de sortie sur lesquelles appuyer son apprentissage. Afin d'identifier ces paramètres, on exploite les formules de calculs de force et de couple qui intègrent, à la fois, les données disponibles et les paramètres que l'on cherche à déterminer. Ainsi, le RN qui accomplit l'IP reçoit en entrée les positions et orientations des particules et estime les forces et les couples qui leurs sont appliqués. Des formules de calcul de ces vecteurs finaux est embarquée dans le RN. On appelle modèle simulateur, illustré Figure III.7, le modèle qui intègre cette partie boîte blanche de transformation des données en sortie du modèle final.

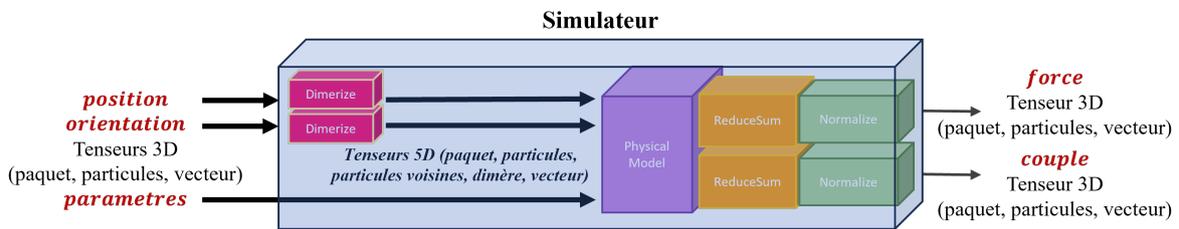


Figure III.7: Modèle simulateur

Il est composé des couches suivantes:

- **Dimerize** dimensionne un tenseur d'entrée pour la couche mathématique. La nouvelle forme du tenseur permet la parallélisation des calculs dans l'UTG. Elle prend un tenseur 3D de la forme (paquet, particules, vecteur) en entrée et le transforme en tenseur 5D (paquet, particules, particules voisines, dimère, vecteur). Son appellation lui vient du terme dimère qui désigne, dans ce cas de la physique, un couple de particules en interaction.
- **PhysicalModel** opère les calculs de force et de couple que les particules exercent les unes sur les autres pour chaque dimère. Les calculs sont différents selon qu'ils proviennent des travaux de Rosenthal ou de l'IRCER. Les formules implémentées dans cette couche sont décrites dans la section III.2.2 pour chacun des cas étudiés.
- **ReduceSum** effectue la somme des interactions entre les paires de particules pour obtenir le vecteur de force final et celui du couple exercés sur chaque particule.
- **Normalize** standardise les résultats de sortie du modèle selon la moyenne et l'écart-type du jeu de données d'entraînement pour la force et le couple. Cette

dernière étape, bien qu'étant infondée sur le plan physique et mathématique du problème, est nécessaire et inhérente à la bonne estimation des pertes d'entraînement et pour l'efficacité de la descente de gradient.

Ce modèle de simulateur est utilisé pour chaque cas d'étude suivant.

III.2.1.3 Le modèle de générateur de paramètres

Le modèle final inclut un générateur des paramètres à identifier en amont du simulateur. La Figure III.8 illustre cette partie pour la formule de calcul du potentiel d'interaction des particules de l'IRCER avec ces cinq paramètres a_1 , a_2 , ϵ , C et λ introduits dans la sous-section III.1.2.2. Afin de réaliser une boîte grise, les paramètres sont des variables entraînaibles du réseau, autrement dit ce sont des poids de cellules neuronales. Contrairement à un réseau composé de plusieurs couches cachées de neurones artificiels, on applique un entraînement à des cellules que l'on peut lire et interpréter. Cependant, c'est donc que le réseau possède extrêmement peu de paramètres entraînaibles, cela occasionne plusieurs défauts d'entraînement qui sont: la disparition et l'explosion des gradients, un temps d'entraînement excessivement long, de l'inertie dans les minima locaux de la courbe des pertes en fonction d'un paramètre du réseau. Pour un problème simple et linéaire, le processus d'AP permet aux valeurs de ces paramètres de fluctuer pour converger jusqu'à stabilisation des pertes d'entraînement du modèle. Or, dans le cas complexe et incertainement linéaire de la thèse, un tel RN dispose de peu de capacité d'apprentissage et identifie, tout en se figeant, des paramètres dont l'estimation est fortement impactée par leurs valeurs initiales.

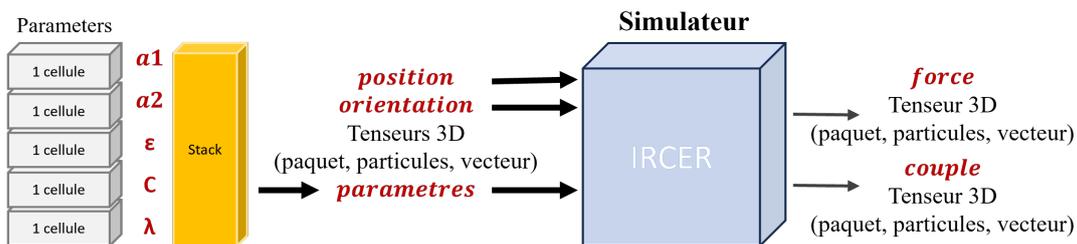


Figure III.8: Modèle en boîte grise pour l'IRCER

Un tel modèle pour cette application soulève des problématiques supplémentaires qui ont trait à l'identification conjointe des paramètres et à l'ordre de grandeur des valeurs de ces derniers. Premièrement, l'identification conjointe des paramètres est un réel problème. Lorsque l'entraînement du modèle évalue son erreur sur l'estimation d'un paramètre cela ne lui apporte pas d'information significative sur la valeur du paramètre si

les autres paramètres qu'il recherche conjointement faussent les résultats. En admettant que la valeur d'un paramètre soit juste, si un autre paramètre ne l'est pas, les résultats des calculs du simulateur ne correspondent pas avec les données d'entraînement alors la fonction de perte soulève une erreur relativement grande qui éclipse une valeur de paramètre juste et la fait varier pour, peut-être, ne plus jamais la retrouver. C'est le phénomène indésirable de l'identification conjointe. Le modèle est victime de l'équation mathématique du modèle simulateur ce qui rend l'apprentissage impossible. L'identification des paramètres est favorisée par une alternance dans l'ajustement de leurs valeurs en figeant les valeurs des autres paramètres et en les entraînant individuellement. Cette stratégie dans l'entraînement atténue le phénomène indésirable sans pour autant l'éliminer. Deuxièmement, on connaît les ordres de grandeurs des valeurs des paramètres impliqués dans les formules, ils ont pour particularité d'être très éloignés les uns des autres, pour être exact on observe une différence de l'ordre d'une puissance approximative de $10e30$ seulement entre ϵ et λ . Cela complique l'ajustement d'un taux d'apprentissage commun entre ces paramètres, voire le rend impossible. Puisque les variations de valeurs pour un petit nombre ne permettent pas de faire varier significativement un grand nombre tandis que la variation d'un grand nombre occasionne des pas immenses dans les variations d'un petit nombre. Les connaissances que l'on possède de ces valeurs de paramètres sont des restrictions bénéfiques qui livrent des informations capitales au RN dans son apprentissage. Les valeurs des bornes présentées en section III.1.2.2 sont utilisées pour personnaliser le modèle au problème d'identification courant tel qu'illustré dans la figure III.9.

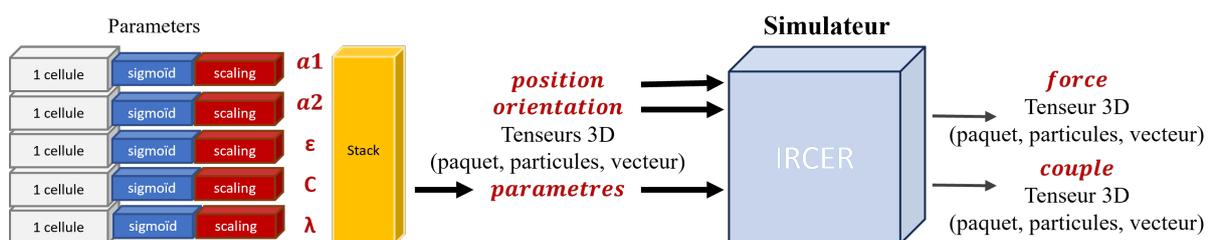


Figure III.9: Modèle de générateur de paramètres pour les formules de l'IRCER

Une fonction sigmoïd permet la normalisation des paramètres entraînaables à des valeurs comprises entre 0 et 1. Une couche, appelée **scaling**, effectue la transformation de la valeur normalisée entre les bornes des valeurs propres des paramètres. Ainsi, l'entraînement peut s'effectuer. Sans incorporer ces étapes dans le modèle, l'entraînement devient instable et ne réduit jamais ses pertes. La résolution de cette seconde problématique de ordres de grandeur permet une variation régulée des valeurs

des paramètres sans générer d'irrégularité dans le résultat de perte d'entraînement. La première problématique d'identification conjointe des paramètres est fortement et favorablement impactée par l'utilisation de ces bornes de valeurs.

Quant au problème exposé en première partie de section concernant le faible nombre de paramètres entraînaibles de ce modèle et ses conséquences, le modèle doit se complexifier pour augmenter le nombre de paramètres entraînaibles, tout en conservant la nature boîte grise de ce dernier et tendre vers l'objectif de cette étude. Pour cela, le modèle suivant, qui est illustré III.10 multiplie le nombre de jeux de paramètres qu'il produit par le nombre de cellules de la couche **Parameter** de façon à multiplier les configurations initiales des valeurs des paramètres ainsi que les minima locaux différents observés par la fonction de perte. Une couche **Evaluator** a autant de cellules que celle des couches **Parameter** qui correspondent aux poids entraînaibles considérés pour chacun des jeux de paramètres afin que le réseau puisse confronter l'influence de ces jeux sur les pertes de l'entraînement et ajuster leur niveau d'implication dans la résolution du problème de régression. Dans l'exemple de la Figure III.10, on compte 32 jeux de paramètres, ce nombre est un hyperparamètre du modèle d'AP que l'on peut configurer pour l'adapter au niveau de complexité du modèle simulateur et au nombre d'échantillon du jeu de données d'entraînement. Plus le problème est complexe, plus on utilise de données, plus le nombre de cellules sera intéressant à augmenter. Le principe qui s'applique à notre cas est en premier lieu un principe fondamental de l'AP.

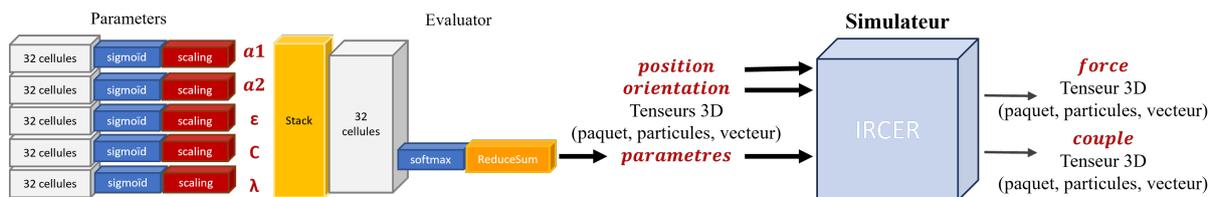


Figure III.10: Modèle final de générateur de paramètres pour les formules de l'IR CER

III.2.2 Tests

III.2.2.1 Preuve de concept

Les chercheurs de l'IRCER supposent qu'il existe un modèle linéaire capable de résoudre leur problème de simulation d'agrégats dans une suspension colloïdale. Si c'est le cas, la méthode d'IP par AP doit aboutir au modèle correspondant sans ambiguïté. Afin de vérifier la faisabilité de cette application, nous avons commencé par étudier le cas d'un système que l'on sait linéaire, pour lequel les particules observées sont isotropes (sans vecteur d'orientation à considérer) et où les paramètres utilisés par la formule sont des constantes connues. L'idée étant d'apporter la preuve que le modèle d'AP peut retrouver les valeurs des paramètres suivants: $\epsilon = 2.0217e-20$, $C = 3.396456e-26$ et $\lambda = 5e6$.

Le jeu de données d'entraînement est du même format que celui décrit en section III.2.1.1 et résulte des simulations des vecteurs de force en fonction des vecteurs de position selon la formule III.7 obtenue à partir de la formule de potentiel d'interaction des particules III.8.

$$\mathbf{F}_{12} = 48\epsilon \frac{\sigma^{12}}{|\mathbf{r}_{12}|^{14}} \mathbf{r}_{12} - 2C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \left(\lambda + \frac{1}{|\mathbf{r}_{12}|} \right) \frac{\mathbf{r}_{12}}{|\mathbf{r}_{12}|} \quad (\text{III.7})$$

$$\phi(\mathbf{r}_{12}) = 4\epsilon \left(\frac{\sigma}{|\mathbf{r}_{12}|} \right)^{12} - 2C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \quad (\text{III.8})$$

D'une part, ce cas sert à expérimenter et valider la méthode simulateur qui embarque des formules de potentiel d'interaction des particules dans un modèle d'AP. D'autre part, il a permis de tester le développement et configurer l'entraînement du modèle final d'IP en vue d'être adapté à un cas de simulation complexe. Les conditions d'expérimentations sont les mêmes que pour le cas d'étude de l'IRCER tout en réduisant la complexité du problème. Simplement, les forces appliquées aux particules sont calculées de la même façon sans considérer les couples associés. Le modèle utilisé, illustré Figure III.11 est une version plus simple du modèle illustré Figure III.10 adapté pour les données du cas de la preuve de concept. Son entraînement est configuré avec un optimiseur RMSprop et utilise la fonction de perte d'erreur moyenne absolue (MAE). Les valeurs des paramètres sont initialisées aléatoirement selon la loi normale de moyenne 0 et de variance 5 (cette valeur correspondant avec les bornes positives et

négatives sur l'axe des abscisses à partir desquelles la fonction sigmoïd s'applatit). La preuve de concept est apportée par les résultats de l'entraînement qui convergent vers les valeurs des paramètres de simulation: $\epsilon = 2.0217e-20$, $C = 3.396456e-26$ et $\lambda = 5e6$.

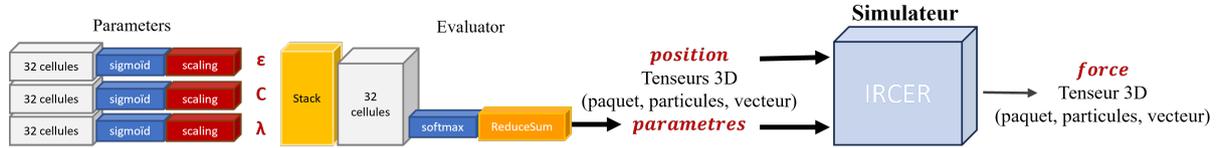


Figure III.11: Modèle pour la preuve de concept

III.2.2.2 Cas d'étude de l'IRCER

Cette section traite le cas qui intéresse Arnaud Videcoq et l'IRCER. Le jeu de données utilisé pour entraîner le modèle d'AP est issu du modèle de simulation actuellement utilisé par l'IRCER pour ces travaux sur les procédés de céramique. Le modèle simulateur embarque les formules complètes de simulations des mouvements des particules présentés dans la section III.1.2.1. Les équations de calcul de force et de couple issue du potentiel d'interaction des particules sont les suivantes: III.9 et III.10 pour le modèle de Rosenthal, III.11 et III.12 pour le modèle de l'IRCER.

$$\mathbf{F}_{12} = 48\epsilon \frac{\sigma^{12}}{|\mathbf{r}_{12}|^{14}} \mathbf{r}_{12} - C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|^2} \left(\left(\lambda + \frac{2}{|\mathbf{r}_{12}|} \right) \frac{(\mathbf{u}_1 - \mathbf{u}_2) \cdot \mathbf{r}_{21}}{|\mathbf{r}_{12}|} \mathbf{r}_{12} + \mathbf{u}_1 - \mathbf{u}_2 \right) \quad (\text{III.9})$$

$$\mathbf{T}_{12} = C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|^2} \mathbf{u}_1 \times \mathbf{r}_{21} \quad (\text{III.10})$$

$$\begin{aligned}
 \mathbf{F}_{12} = & 48\epsilon \frac{\sigma^{12}}{|\mathbf{r}_{12}|^{14}} \mathbf{r}_{12} + C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \left(\lambda + \frac{1}{|\mathbf{r}_{12}|} \right) \\
 & \left(a_1 \left(\mathbf{u}_1 \frac{\mathbf{r}_{21}}{|\mathbf{r}_{21}|} \mathbf{u}_2 \frac{\mathbf{r}_{21}}{|\mathbf{r}_{21}|} \right) + \left(a_2 - \frac{a_1}{3} \right) \mathbf{u}_1 \mathbf{u}_2 \right) \frac{\mathbf{r}_{12}}{|\mathbf{r}_{12}|} \\
 & + C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \\
 & \left(-\frac{2a_1 \mathbf{u}_1 \mathbf{r}_{21} \mathbf{u}_2 \mathbf{r}_{21}}{|\mathbf{r}_{12}|^4} \mathbf{r}_{21} + \frac{a_1 (\mathbf{u}_1 (\mathbf{u}_2 \mathbf{r}_{21})) + (\mathbf{u}_2 (\mathbf{u}_1 \mathbf{r}_{21}))}{|\mathbf{r}_{12}|^2} \right)
 \end{aligned} \tag{III.11}$$

$$\begin{aligned}
 \mathbf{T}_{12} = & -a_1 \mathbf{u}_2 \mathbf{r}_{21} C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|^3} \mathbf{u}_1 \times \mathbf{r}_{21} \\
 & + \left(a_2 - \frac{a_1}{3} \right) C \frac{\exp(-\lambda(|\mathbf{r}_{12}| - \sigma))}{|\mathbf{r}_{12}|} \mathbf{u}_2 \times \mathbf{u}_1
 \end{aligned} \tag{III.12}$$

Contrairement au cas de la preuve de concept, les premières tentatives d'entraînement du modèle ont révélé son incapacité à converger vers des valeurs constantes de paramètres qui permettent de calculer les mêmes résultats que les données d'entraînement de force et de couple pour la globalité des données et des temps de simulation. On suppose qu'il existe un jeu de paramètres localement plus adapté pour tendre vers les résultats de simulations. C'est à dire qu'un jeu de paramètres pourrait être différent d'un temps de simulation à l'autre, d'un fichier de données à l'autre, d'une interaction à l'autre. C'est le problème que pose la littérature en matière de calcul de potentiel d'interaction des particules aujourd'hui : les paramètres varient d'une configuration de paire de particules à l'autre. C'est pourquoi, nous proposons un entraînement dit "local" des jeux de paramètres par le moyen du modèle illustré Figure III.12 tout en gardant pour objectif d'identifier des paramètres globaux qui simulent de manière convaincante les mouvements des particules.

Ce modèle prend en entrée un tenseur de la taille des cellules des couches du réseau qui détient la valeur de l'indice du jeu de paramètres à utiliser pour les formules de calculs du simulateur pour chaque échantillon. De cette façon, un jeu de paramètres n'apprendra pas du jeu de données d'entraînement mais d'une fraction de celui-ci dont le volume dépend du rapport $\gamma = \frac{\text{cellules}}{\text{échantillons}}$. C'est ce que l'on appelle un entraînement local, par opposition à l'entraînement global.

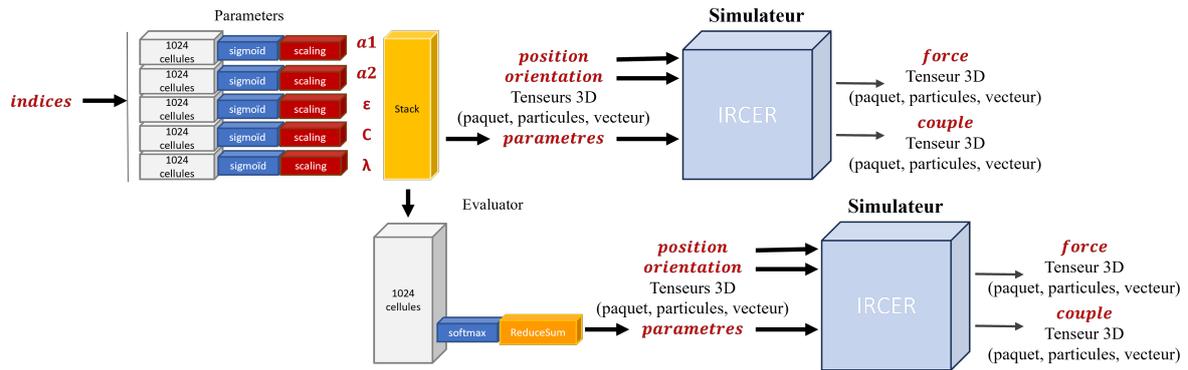


Figure III.12: Modèle de générateur de paramètre avec entraînement local

Les bornes des valeurs des paramètres sont un point clef de la réussite de l'entraînement du modèle, ils apportent une solution efficace aux problèmes d'identification conjointe des paramètres et à l'écart d'ordres de grandeur de leurs valeurs. Ces contraintes peuvent être ré-estimées sur la base des valeurs de jeux de paramètres entraînés. Les minimums et maximums de leurs valeurs servent à mettre à jour les bornes pour affiner la recherche des paramètres. Afin d'éviter les valeurs aberrantes, communément appelées outliers en anglais, et d'empêcher l'inertie, la standardisation des paramètres permet d'identifier les nouvelles bornes aux valeurs -3 et 3. Cette technique de rafraîchissement des bornes ouvre aussi la perspective d'élargissement des bornes initiales pour explorer de nouvelles possibilités.

Le taux d'apprentissage de l'optimiseur définit le pas appliqué à la descente de gradient sur chaque courbe de perte des paramètres du réseau. Plus il sera élevé plus le modèle apprendra rapidement mais il ne pourra pas trouver le minimum global d'une courbe de perte si sa valeur ne figure pas à son échelle. Plus il sera bas plus le modèle mettra de temps à apprendre des données et plus il sera victime d'inertie dans les minima locaux d'une courbe de perte. Il faut ajuster le taux d'apprentissage afin qu'il ne soit ni trop grand ni trop petit pour la courbe de perte du problème à résoudre. La normalisation des données est un pré-traitement qui aide considérablement à fixer la valeur de taux d'apprentissage pour éviter de tenir compte d'ordres de grandeurs extrêmes sur une courbe de perte. L'optimiseur RMSprop présente des avantages à gérer nativement le pourrissement du taux d'apprentissage pour stabiliser l'entraînement après de nombreuses époques et, par le biais d'hyperparamètres et d'algorithmes propres, à favoriser la recherche du minimum global indépendamment du taux d'apprentissage. L'identification conjointe des paramètres rend l'entraînement particulièrement sensibles aux minima de perte locaux. Alors, on porte une attention particulière à cet

hyperparamètre pour observer son influence sur l'IP.

La couche **Evaluator** participe également à pondérer les minima locaux supposément apparus dans les jeux de paramètres générés pour converger vers le meilleur d'entre eux si ce n'est le minimum global. Pour l'aider dans cette tâche, un bruit gaussien peut être paramétré pour généraliser l'entraînement des jeux de paramètres, il est ajouté aux poids de l'**Evaluator**.

La section suivante présente les résultats obtenus pour différentes configurations d'entraînement et pour les modèles de Rosenthal et de l'IRCER.

III.2.2.3 Analyse des résultats

On commence par évaluer l'influence de chaque hyperparamètre qu'offre notre modèle sur la résolution du problème. Le taux d'apprentissage est noté α et la perte globale de l'entraînement est notée P_g (pour rappel: nous utilisons la fonction MAE), ce résultat correspond à la moyenne des valeurs absolues des différences entre les forces et couples normalisés (selon une moyenne respective nulle et un écart-type de 1) d'entraînement et de sortie du RN. Les bornes des valeurs des paramètres entraînaibles sont celles définies dans la section III.1.2.2.

Cellules	Bornes	α	Entraînement local	Bruit	P_g	Vitesse
1	∅	0.001	∅	∅	0.428	Lente
1	∅	0.1 ~ 0.0005	∅	∅	0.428	Lente
32	figées	0.001	sans	0	0.427	Moyenne
32	figées	0.1 ~ 0.0005	sans	0	0.427	Rapide
32	figées	0.1 ~ 0.0001	sans	0	0.427	Moyenne
64	figées	0.001	sans	0	0.427	Moyenne
64	figées	0.1 ~ 0.0005	sans	0	0.427	Rapide
256	figées	0.001	sans	0	0.427	Moyenne
256	figées	0.1 ~ 0.0005	sans	0	0.427	Rapide
1024	figées	0.001	sans	0	0.427	Moyenne
1024	figées	0.1 ~ 0.0005	sans	0	0.427	Rapide

Table III.3: Résultats d'entraînement par nombre de cellules pour le modèle IRCER

Le tableau III.3 montre qu'un petit nombre de cellules, donc de jeux de paramètres entraînaibles, suffit à améliorer la recherche des paramètres tout en accélérant la

vitesse relative d'apprentissage. Le pourrissement exponentiel du taux d'apprentissage n'impacte pas le score de perte cependant il permet également à l'entraînement d'être plus rapide. Par la suite, nous utilisons régulièrement le pourrissement du taux d'apprentissage et 32 cellules pour parvenir à nos pertes minimums.

Cellules	Bornes	α	Entraînement local	Bruit	P_g
32	figées	0.1 ~ 0.0005	sans	0	0.427
32	figées	0.1 ~ 0.0005	sans	0.05	0.427
32	figées	0.1 ~ 0.0005	sans	0.15	0.428
32	figées	0.1 ~ 0.0005	sans	0.30	0.428

Table III.4: Résultats d'entraînement par bruit de l'Evaluator pour le modèle IRCER

Le bruit dans l'Evaluator stimule l'apprentissage de toutes les cellules du modèle pour éviter l'inertie dans les minimas locaux. Il apparaît que dans ce cas, voir tableau III.4, il n'influence pas la résolution du problème et le complique lorsqu'il est trop élevé.

Cellules	Bornes	α	Entraînement local	Bruit	P_g
32	figées	0.1 ~ 0.0005	sans	0	0.427
32	figées	0.1 ~ 0.0005	avec	0	0.427
32	mobiles	0.1 ~ 0.0005	avec	0	0.427
256	mobiles	0.1 ~ 0.0005	avec	0.05	0.427
32	larges et figées	0.1 ~ 0.0005	sans	0	0.424
32	larges et figées	0.1 ~ 0.0005	avec et figé	0.05	0.424
32	larges et figées	0.1 ~ 0.0005	avec	0	0.424
1024	larges et figées	0.1 ~ 0.0005	avec et figé	0.05	0.427
1024	larges et figées	0.1 ~ 0.0005	avec	0.05	0.427
64	très larges et figées	0.1 ~ 0.0005	sans	0	0.425
1024	très larges et mobiles	0.1 ~ 0.0005	avec	0	0.426

Table III.5: Résultats d'apprentissage par configuration d'entraînement pour le modèle IRCER

Le tableau III.5 montre que l'entraînement local des jeux de paramètres, qu'il soit figé avant l'entraînement global ou non, n'influence visiblement pas le score de perte final. Un entraînement local figé signifie que les valeurs des paramètres entraînaibles sont figés lorsque les poids de la couche Evaluator sont entraînés. La mise à jour des valeurs des bornes des paramètres, désigné par le terme "bornes mobiles", tel que

décrit dans la section III.2.2.2 n'aide pas non plus l'apprentissage. En revanche, un élargissement des bornes a rendu le modèle plus performant. Or, au delà des bornes larges suivantes: $k_B T \sigma < C < 15 k_B T \sigma$, $k_B T < \epsilon < 10 k_B T$, $\frac{1}{\sigma} < \lambda < \frac{10}{\sigma}$, $-10 < a_1 < 0$ et $-5 < a_2 < 5$, le modèle régresse.

Ces observations valent pour les formules de calculs et le jeu de données de simulations de l'IRCER. Il n'est pas établi que l'influence de ces hyperparamètres du modèle ne change pas pour des calculs d'interactions de particules différents. Le tableau III.6 montre les résultats de l'IP réalisée par le modèle III.10 en confrontant les formules de Rosenthal à celles de l'IRCER pour les différents temps de simulation de structures colloïdales. Les symboles e_F et e_C sont respectivement les valeurs des erreurs moyennes d'estimation du modèle sur le calcul des forces et des couples des échantillons d'interactions de particules de tests du modèle (c'est-à-dire, des données qui n'ont pas servies à entraîner le modèle). La configuration d'entraînement découle de nos observations précédentes, soit que le nombre de cellules est 32, le taux d'apprentissage pourrait exponentiellement entre 0.1 et 0.0005, il n'y a pas de bruit dans l'Evaluator ni d'entraînement local. Les bornes des valeurs des paramètres ne sont pas élargies au-delà de celles établies par l'IRCER dans la section III.1.2.2 pour rester dans le cadre du projet initial.

La première constatation est qu'aucun des deux modèles de simulation de structures colloïdales ne permet de reproduire le jeu de données de simulation de l'IRCER avec des paramètres de simulation constants. Toutefois, les formules de l'IRCER présentent moins d'erreurs d'estimation que celles de Rosenthal. Il est intéressant de voir que certains paramètres identifiés varient au cours du temps de simulation et d'autres non. La force de couplage C semble unanimement et relativement constante au cours du temps, ce qui signifie que, dans le cadre de notre modélisation, l'intensité de l'interaction des particules n'est pas influencée par la formation d'agrégats de particules. Il en est de même pour λ qui caractérise la portée de cette interaction. Le paramètre ϵ quant à lui subit une augmentation à partir de la première seconde de simulation. Ce changement est cohérent puisqu'il empêche le chevauchement théorique des particules, il semble pertinent d'imaginer que l'intensité de répulsion à courte distance augmente lorsque les particules s'agrègent. La présence de poids tels qu' a_1 et a_2 apporte une meilleure caractérisation des interactions des particules. Tandis qu' a_1 est presque strictement constant, a_2 pondère variablement les interactions au cours du temps en suivant une tendance à l'augmentation au cours de la simulation. Pour le travail d'IP sur l'ensemble des données, les valeurs trouvées pour les formules de l'IRCER sont

Modèle	$t(s)$	ϵ	C	λ	a_1	a_2	e_F	e_C	P_g
IRCER	0.01	$8.0916e^{-21}$	$7.285e^{-27}$	$8.3315e^6$	-1.0009	0.0903	$3.5986e^{-14}$	$1.0939e^{-21}$	0.1029
Rosenthal		$1.0308e^{-20}$	$7.9497e^{-27}$	$8.3325e^6$			$4.463e^{-14}$	$1.0939e^{-21}$	0.1109
IRCER	0.1	$8.0869e^{-21}$	$7.2783e^{-27}$	$8.3331e^6$	-1.0001	0.1052	$1.7926e^{-13}$	$4.9622e^{-21}$	0.2077
Rosenthal		$8.0869e^{-21}$	$7.2782e^{-27}$	$8.3333e^6$			$2.0916e^{-13}$	$4.9622e^{-21}$	0.2158
IRCER	0.2	$8.087e^{-21}$	$7.2787e^{-27}$	$8.3316e^6$	-1.0001	0.0807	$2.5434e^{-13}$	$8.4995e^{-21}$	0.2511
Rosenthal		$8.0869e^{-21}$	$7.2782e^{-27}$	$8.3331e^6$			$2.97826e^{-13}$	$8.4995e^{-21}$	0.2579
IRCER	0.3	$8.0869e^{-21}$	$7.2786e^{-27}$	$8.3329e^6$	-1.0001	0.0805	$4.1117e^{-13}$	$1.2245e^{-20}$	0.2839
Rosenthal		$8.0869e^{-21}$	$7.2782e^{-27}$	$8.333e^6$			$4.6835e^{-13}$	$1.2245e^{-20}$	0.2911
IRCER	0.4	$8.0869e^{-21}$	$7.2782e^{-27}$	$8.3331e^6$	-1	0.0966	$4.8536e^{-13}$	$1.5729e^{-20}$	0.3144
Rosenthal		$8.0872e^{-21}$	$7.2786e^{-27}$	$8.3326e^6$			$5.5546e^{-13}$	$1.5729e^{-20}$	0.3194
IRCER	0.5	$8.088e^{-21}$	$7.2808e^{-27}$	$8.3281e^6$	-1.0003	0.1357	$6.926e^{-13}$	$2.2143e^{-20}$	0.332
Rosenthal		$8.088e^{-21}$	$7.2793e^{-27}$	$8.3299e^6$			$7.562e^{-13}$	$2.2143e^{-20}$	0.3364
IRCER	0.6	$8.0873e^{-21}$	$7.2789e^{-27}$	$8.3321e^6$	-1.0001	0.1196	$7.427e^{-13}$	$2.3166e^{-20}$	0.3531
Rosenthal		$8.0881e^{-21}$	$7.2793e^{-27}$	$8.3296e^6$			$8.1494e^{-13}$	$2.3166e^{-20}$	0.3576
IRCER	0.7	$8.0874e^{-21}$	$7.2791e^{-27}$	$8.3315e^6$	-1.0001	0.1445	$9.6e^{-13}$	$3.1064e^{-20}$	0.3692
Rosenthal		$8.0883e^{-21}$	$7.2795e^{-27}$	$8.3293e^6$			$1.0255e^{-12}$	$3.1064e^{-20}$	0.3725
IRCER	0.8	$8.0875e^{-21}$	$7.279e^{-27}$	$8.3315e^6$	-1.0001	0.1353	$1.0907e^{-12}$	$3.3551e^{-20}$	0.3781
Rosenthal		$8.0878e^{-21}$	$7.279e^{-27}$	$8.3303e^6$			$1.172e^{-12}$	$3.3551e^{-20}$	0.3812
IRCER	0.9	$8.0877e^{-21}$	$7.279e^{-27}$	$8.3317e^6$	-1.0001	0.165	$1.214e^{-12}$	$3.976e^{-20}$	0.4003
Rosenthal		$8.0884e^{-21}$	$7.2793e^{-27}$	$8.3297e^6$			$1.283e^{-12}$	$3.976e^{-20}$	0.4035
IRCER	0 ~ 1	$8.0868e^{-21}$	$7.2781e^{-27}$	$8.3333e^6$	-1	0.14	$6.049e^{-13}$	$1.9113e^{-20}$	0.3054
Rosenthal		$8.0868e^{-21}$	$7.2781e^{-27}$	$8.3333e^6$			$6.5893e^{-13}$	$1.9113e^{-20}$	0.3094
IRCER	1	$8.09e^{-21}$	$7.2817e^{-27}$	$8.3269e^6$	-1.0004	0.211	$1.3125e^{-12}$	$4.21e^{-20}$	0.4131
Rosenthal		$8.0899e^{-21}$	$7.28e^{-27}$	$8.328e^6$			$1.3823e^{-12}$	$4.21e^{-20}$	0.4162
IRCER	2	$9.7273e^{-21}$	$7.2884e^{-27}$	$8.3182e^6$	-1.0013	0.2854	$2.3459e^{-12}$	$6.7527e^{-20}$	0.4971
Rosenthal		$9.8523e^{-21}$	$7.2791e^{-27}$	$8.3313e^6$			$2.418e^{-12}$	$6.7527e^{-20}$	0.4997
IRCER	3	$9.9484e^{-21}$	$7.2817e^{-27}$	$8.3301e^6$	-1.0004	0.3654	$2.6811e^{-12}$	$7.4124e^{-20}$	0.5173
Rosenthal		$1.0247e^{-20}$	$7.2797e^{-27}$	$8.3299e^6$			$2.7469e^{-12}$	$7.4124e^{-20}$	0.5202
IRCER	4	$1.0581e^{-20}$	$7.2847e^{-27}$	$8.3263e^6$	-1.0008	0.333	$2.9159e^{-12}$	$7.9113e^{-20}$	0.5307
Rosenthal		$1.0837e^{-20}$	$7.28e^{-27}$	$8.331e^6$			$2.9929e^{-12}$	$7.9113e^{-20}$	0.5339
IRCER	5	$1.0396e^{-20}$	$7.2853e^{-27}$	$8.3261e^6$	-1.0007	0.3970	$2.972e^{-12}$	$8.0637e^{-20}$	0.5314
Rosenthal		$1.0786e^{-20}$	$7.2796e^{-27}$	$8.3308e^6$			$3.0372e^{-12}$	$8.0637e^{-20}$	0.5354
IRCER	6	$1.0549e^{-20}$	$7.2918e^{-27}$	$8.3194e^6$	-1.0016	0.4283	$3.02e^{-12}$	$7.6892e^{-20}$	0.5352
Rosenthal		$1.1035e^{-20}$	$7.28e^{-27}$	$8.3301e^6$			$3.078e^{-12}$	$7.6892e^{-20}$	0.5383
IRCER	7	$1.05e^{-20}$	$7.281e^{-27}$	$8.3287e^6$	-1.0003	0.4785	$3.241e^{-12}$	$8.6915e^{-20}$	0.554
Rosenthal		$1.1111e^{-20}$	$7.2787e^{-27}$	$8.3324e^6$			$3.3077e^{-12}$	$8.6915e^{-20}$	0.5592
IRCER	8	$1.0142e^{-20}$	$7.2812e^{-27}$	$8.3284e^6$	-1.0004	0.5832	$3.2554e^{-12}$	$8.7453e^{-20}$	0.548
Rosenthal		$1.0802e^{-20}$	$7.2788e^{-27}$	$8.332e^6$			$3.3172e^{-12}$	$8.7453e^{-20}$	0.5524
IRCER	9	$1.0065e^{-20}$	$7.2859e^{-27}$	$8.3256e^6$	-1.0009	0.6773	$3.5817e^{-12}$	$8.9264e^{-20}$	0.564
Rosenthal		$1.0931e^{-20}$	$7.2801e^{-27}$	$8.3299e^6$			$3.621e^{-12}$	$8.9264e^{-20}$	0.5682
IRCER	10	$1.0443e^{-20}$	$7.2795e^{-27}$	$8.3319e^6$	-1.0002	0.6479	$3.529e^{-12}$	$8.8188e^{-20}$	0.5639
Rosenthal		$1.1291e^{-20}$	$7.2806e^{-27}$	$8.3294e^6$			$3.5712e^{-12}$	$8.8188e^{-20}$	0.5674
IRCER	1 ~ 10	$1.0001e^{-20}$	$7.2781e^{-27}$	$8.3333e^6$	-1	0.4382	$2.856e^{-12}$	$7.7e^{-20}$	0.5265
Rosenthal		$1.0512e^{-20}$	$7.2781e^{-27}$	$8.3333e^6$			$2.9197e^{-12}$	$7.7e^{-20}$	0.5299
IRCER	0 ~ 10	$8.0869e^{-21}$	$7.2781e^{-27}$	$8.3333e^6$	-1	0.33	$1.7471e^{-12}$	$4.7852e^{-20}$	0.427
Rosenthal		$8.0915e^{-21}$	$7.2781e^{-27}$	$8.3333e^6$			$1.8057e^{-12}$	$4.7852e^{-20}$	0.4302

Table III.6: Résultats d'apprentissage par temps de simulation

donc : $\epsilon = 8.0869e^{-21}$, $C = 7.2781e^{-27}$, $\lambda = 8.3333e^6$, $a_1 = -1$ et $a_2 = 0.33$.

L'amplitude des valeurs de force et de couple sur l'ensemble du jeu de données vaut respectivement: $1.0812e^{-10}$ et $3.1969e^{-18}$. Les erreurs moyennes de tests du modèle identifié de l'IRCER sont de $1.7471e^{-12}$ pour la force et $4.7852e^{-20}$ pour le couple. Cela signifie que le modèle présente une erreur moyenne relative d'estimation de la force d'environ 1.616% et d'environ 1.497% du couple.

Le tableau suivant III.7 est une étude de l'entraînement local. Avec les précédents résultats, on a observé les paramètres entraînaibles varier au cours du temps de simulation des colloïdes. On veut observer les variations des paramètres en fonction des données. Alors, on augmente le nombre de cellules jusqu'à ce qu'il soit équivalent au nombre d'échantillons de données pour qu'il existe un jeu de paramètres dédié à un résultat de force et de couple. Pour rappel, γ est le rapport $\frac{\text{cellules}}{\text{echantillons}}$. On considère les données à 0,4 secondes de temps de simulation. Sur les 1000 échantillons existants, 16 ont été retirés de l'entraînement pour être utile à la validation. La perte locale de l'entraînement est notée P_l .

Cellules	γ	ϵ	C	λ	a_1	a_2	P_l	P_g
1	0%	$8.0868e^{-21}$	$7.2781e^{-27}$	$8.3333e^6$	-1	0.0967	0.3144	0.3144
32	3%	$8.0869e^{-21}$	$7.2782e^{-27}$	$8.3331e^6$	-1	0.0966	0.3436	0.31435
246	25%	$8.0868e^{-21}$	$7.2782e^{-27}$	$8.3332e^6$	-1	0.0967	0.3166	0.31435
492	50%	$8.0868e^{-21}$	$7.2782e^{-27}$	$8.333e^6$	-1	0.0966	0.31535	0.3144
984	100%	$8.0868e^{-21}$	$7.2782e^{-27}$	$8.3329e^6$	-1	0.0966	0.31465	0.31435

Table III.7: Résultats d'apprentissage pour différents volumes de neurones à 0.4 secondes de temps de simulation

Comme précédemment évoqué, l'entraînement local n'a pas amélioré les performances globales du modèle. Or, on constate que les pertes locales, elles, ont diminué au fur et à mesure de l'augmentation du nombre de cellules. Pour autant, elles n'ont pas diminué au delà des pertes globales et sont aussi plus importantes lorsque le nombre de cellules est bas. Deux stratégies d'apprentissage entrent en conflit : la quantité de données utilisée pour l'entraînement et la spécialisation d'un jeu de paramètres à un résultat spécifique. On en déduit ce qui suit :

- L'apprentissage est croissant en fonction du nombre de données utilisées et représentatives du problème à résoudre, ce qui est un principe même de l'AP.

Lorsque γ est inférieur à 100%, l'entraînement global bénéficie de plus de données d'entraînement que celui qui est local et généralise mieux les paramètres.

- L'avantage de l'entraînement local à se spécifier à une somme d'interactions particulière est compensé lorsque le nombre d'interactions qu'il considère augmente ($\gamma < 100\%$)
- Les formules de calculs ne sont pas adaptées pour reproduire exactement les données de simulation puisque la perte locale ne progresse pas au delà de la perte globale lorsque γ augmente.

Au regard de ces résultats et des précédents, l'entraînement local présente peu d'intérêt pour l'entraînement du modèle et l'IP si les formules de calculs de force et de couple ne peuvent pas aboutir à un seul résultat avec un jeu de paramètres dédiée. C'est une équation insoluble.

III.3 Bilan

Ces travaux sont les premiers d'AP à être menés en collaboration avec l'équipe de l'axe Procédés céramiques de l'IRCER. Pour poursuivre leurs études autour de la modélisation des interactions des particules Janus en suspension colloïdale, l'IP des paramètres de leurs modèles était une piste à explorer qui s'inscrit dans le contexte scientifique actuel qui multiplie les applications d'AP dans tous les domaines. Ce chapitre est une étude technique et exploratrice de modèles d'AP qui permettent d'accomplir la tâche confiée par l'IRCER à XLIM. Il s'agit d'une contribution à la littérature en matière d'IP par AP et au domaine de la céramique. Pour la suite à court terme, Arnaud Videcoq et son équipe doivent déterminer si les valeurs des paramètres estimées ici offrent un nouveau point de vue sur les formules étudiées. Toutefois, pour la partie automatique des travaux, les résultats confirment que la formule de calcul de potentiel d'interaction de particules Janus mise au point à l'IRCER décrit plus fidèlement les colloïdes de céramique que celle de Rosenthal.

Les perspectives de ces travaux sont les suivantes:

- Implémenter plus de formules de potentiel d'interaction de particules Janus en suspension dans nos modèles d'IP, comme par exemple celles de DeLaCruz [105]. Il serait intéressant de confronter les résultats d'estimation de force et de couple et de pertes d'entraînement de plusieurs formules issues de la littérature pour les données de l'IRCER et pas seulement. Notre méthode d'IP peut rechercher les systèmes les mieux adaptés à n'importe quelle application d'interactions de particules.
- Simuler de nouvelles données de suspensions colloïdales au sein de l'IRCER en exploitant les résultats de paramètres apportés par la thèse. Observer les formations d'agrégats de particules dans ces nouvelles simulations tout en les comparant avec des données expérimentales permettrait d'évaluer concrètement la contribution de ces travaux de thèse aux procédés céramiques.
- Utiliser les résultats pour affiner le modèle analytique sous forme d'équations non linéaires.

Conclusion générale

Sommaire

Bilan général	146
Contributions	147
Perspectives	148

Bilan général

Le projet de thèse initial portait l'ambition d'un couple d'entités coopérant pour l'acquisition et le traitement de données. Les travaux menés pendant ces trois années de thèse se sont affranchis de ce concept pour se concentrer sur la partie traitement des données. Ce choix de direction a été motivé par les possibilités de traitements différents offertes par l'AP qui méritaient d'être largement explorées et approfondies. Pour autant le travail effectué ouvre des perspectives pour la partie acquisition. Dans le premier contexte d'application, une théorie construite et introduite dans le chapitre II, section II.1.1.1 propose une optimisation de matrice d'acquisition de données au niveau des noeuds de capteurs basée sur les résultats et méthodes développées dans la thèse. Dans le second contexte d'application, l'exploitation des résultats permet directement l'emploi d'un modèle de simulation, identifié par le travail réalisé, pour l'acquisition des données.

L'AP est un outil de traitement, de transformation, d'analyse, de prédiction et de classification de données récent et en développement. Son usage intéresse les chercheurs et les industriels dans tous les domaines. Le nombre de publications et d'applications de modèles de RNs augmente considérablement depuis que les performances de calculs et de mémoire du matériel informatique permet leur implémentation. Dans ce contexte, l'état-de-l'art évolue d'années en années, si ce n'est de mois en mois. L'étude de la littérature menée dans cette thèse sera probablement obsolète en 2025. Les Transformeurs et les modèles de diffusion semblent des modèles prometteurs pour des applications innovantes dans tous les domaines. Ces modèles et leurs variantes tendent à dominer la littérature pour la décennie à venir. Néanmoins, tant que les Transformeurs resteront aussi gourmands en ressources, ils ne pourront pas rendre obsolètes les applications à bases de cellules récurrentes LSTM et GRU. D'un autre côté, les AEs convolutifs représentent une garantie de performances dans presque tous les domaines non séquentiels.

Les deux contextes d'application des travaux de thèse et leurs problématiques diffèrent fortement bien que l'on puisse considérer que l'on traite des jeux de données spatio-temporelles dans les deux cas. La difficulté majeure dans l'AP aujourd'hui, on le voit bien dans la littérature, est de proposer des modèles de traitement des données flexibles et généralisables. En fin de compte, nous avons constaté qu'une méthode d'AP tire le meilleur de ses performances de sa rigidité, lorsqu'elle est spécifiquement dédiée à une application. Les différentes études enrichissent le domaine de l'AP et chaque nouvelle méthode ou nouveau modèle apporté à l'état-de-l'art contribue à rendre possible de

nouvelles applications inspirées par les précédentes. Dans les domaines respectifs des RCSFs et des suspensions colloïdales de céramique, nous avons contribué et innové en répondant à des problématiques actuelles.

Contributions

Les contributions principales de la thèse sont :

- Le développement de méthodes à l'état de l'art d'AP temporelle, spatiale et spatio-temporelle pour la reconstruction de signaux hautement comprimés.

La thèse présente une méthodologie complète qui permet d'implémenter plusieurs modèles de RNs pour des applications de reconstruction temporelle, spatiale et spatio-temporelle de données raster ou de STs. Notre approche modulaire sert l'utilisateur des méthodes en lui proposant divers outils et modes de reconstruction dont l'objectif est de couvrir un bon nombre de cas différents de besoins et de contraintes, sur le plan des données disponibles, de leur format, du matériel et des critères de restauration.

- La conception d'un modèle de générateur de paramètres ajustables aux formules de calculs de potentiel d'interactions entre particules.

La réalisation d'un tel modèle est basée sur un projet initial clair : employer la puissance de résolution de problème de la méthode de descente de gradients de l'AP à l'identification de système. La preuve de concept établie dans ces travaux montre de manière concluante et concrète l'accomplissement de ce projet. Indépendamment des résultats apportés par la mise en oeuvre de notre méthode aux systèmes colloïdaux de l'IRCER, ce modèle de générateur de paramètres est prêt à intégrer divers données, formules et paramètres en tant que modèle d'identification paramétrique à part entière.

Les contributions secondaires de la thèse sont :

- Une exploration et une analyse complète de la littérature sur l'AP dédiée à la reconstruction de séries temporelles et spatio-temporelles.
- La création d'une IPA open source dédiée aux solutions d'AP pour la reconstruction de signaux hautement comprimés.

- La conception d'un modèle de discriminateur capable d'automatiser la maintenance de l'apprentissage des réseaux de neurones impliqués dans les méthodes de la thèse.
- La conception d'un modèle de générateur de paramètres capable de discréditer les formules de calculs de potentiel d'interactions de particules.
- Le développement d'algorithmes de post-traitement d'optimisation de reconstruction de séries temporelles.

Les résultats qui ressortent du chapitre II dans le domaine des RCSFs remplissent nos objectifs et surpassent nos attentes initiales. La première publication de ces travaux à la 12ème conférence IEEE IDAACS est une contribution à la littérature et nous prévoyons de soumettre une nouvelle publication à propos de notre méthode de reconstruction spatio-temporelle au journal IEEE "Transactions on Signal and Information Processing over Networks".

Bien que les objectifs initiaux des travaux en collaboration avec l'IRCER n'aient pas été atteints. L'étude du chapitre III et ces résultats sont prometteurs pour l'avenir des simulations de mouvement brownien assistées par IA et peuvent ouvrir la voie à de futures recherches sur ces thématiques au sein du LabEx Σ -LIM.

Perspectives

Les chapitres II et III de cette thèse abordent dans leur bilan respectif l'ensemble des perspectives relatives à chacun (cf II.3 et III.3). Pour ce qui concerne les intérêts du projet de thèse et des projets descendants, les perspectives conséquentes à court et moyens termes sont plus spécifiquement les suivantes :

- La co-conception d'un système d'acquisition et de reconstruction des signaux compressés des objets connectés est au coeur des intentions de recherche. Les prochains travaux devront être menés sur le plan de l'acquisition par AP dans une démarche collaborative avec les méthodes de reconstruction, de telle sorte qu'une logique commune de restauration des signaux permette de les observer avec la plus grande parcimonie. Bien entendu, cette perspective s'inscrit dans les objectifs de la thèse sur le plan des enjeux économique et écologique en réduisant les coûts de fonctionnement des objets connectés, dont les RCSFs.

- D'un côté, l'identification paramétrique par AP dédié aux formules de calculs d'interaction de particules Janus en suspension est une réussite sur le plan théorique et expérimental. D'un autre côté, les résultats obtenus sur le plan applicatif ne sont pas satisfaisants. Les paramètres identifiés ne conviennent pas à décrire les formations d'agrégats dans les suspensions colloïdales de l'IRCER par le biais des formules pourvues. C'est pourquoi un travail de recherche doit se concentrer sur une nouvelle formulation des modèles de calculs ou d'estimation des paramètres ou d'utilisation des données de simulations dans le but d'identifier un système dynamique qui correspondra aux mesures expérimentales et numériques de l'IRCER.



Annexes

Sommaire

A.1	API deepsignals	151
A.1.1	Diagramme de paquetages	151
A.1.2	Diagramme de classes	152
A.1.3	Diagramme de cas d'utilisation	154
A.2	Jeu de données	156
A.2.1	Signaux d'un réseau de capteur sans fil	156

A.1 API deepsignals

A.1.1 Diagramme de paquetages

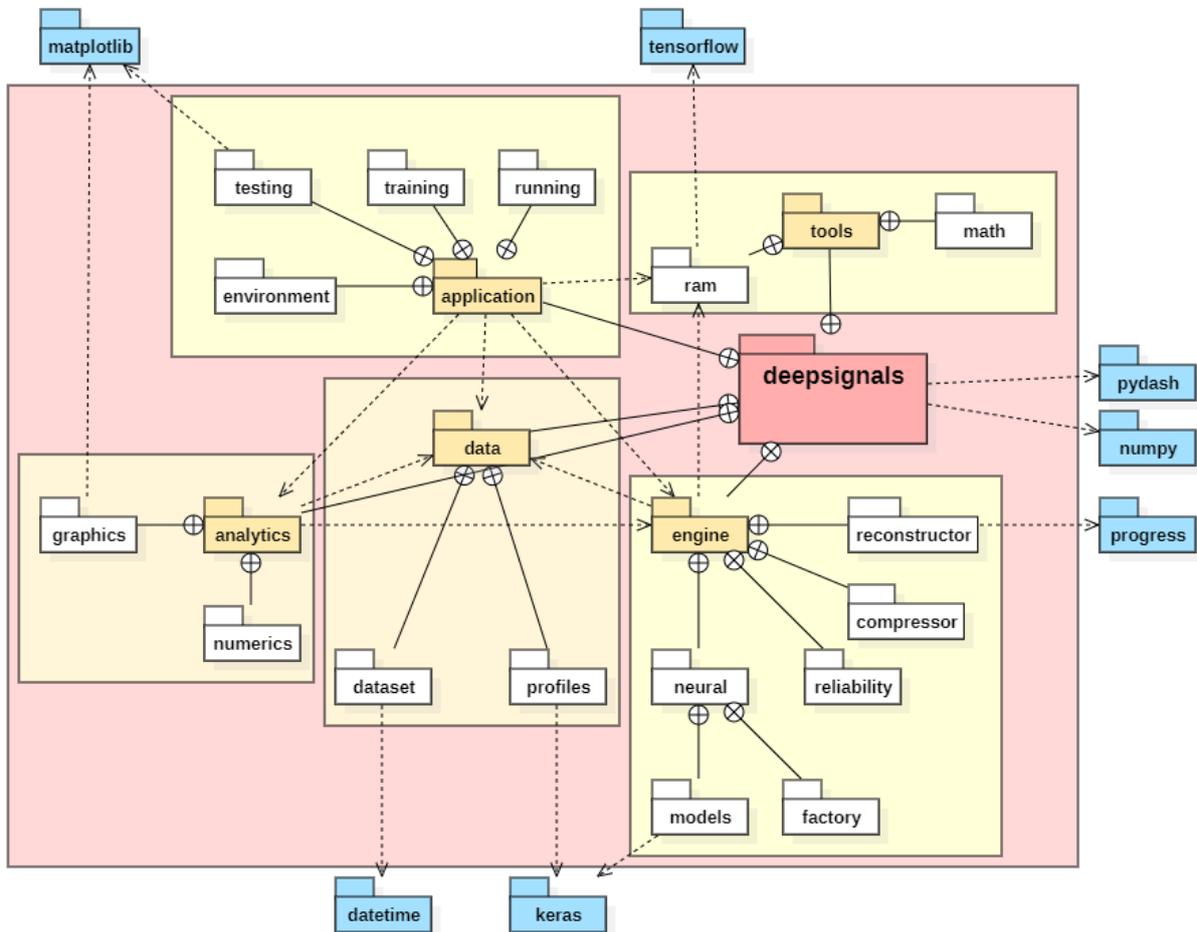
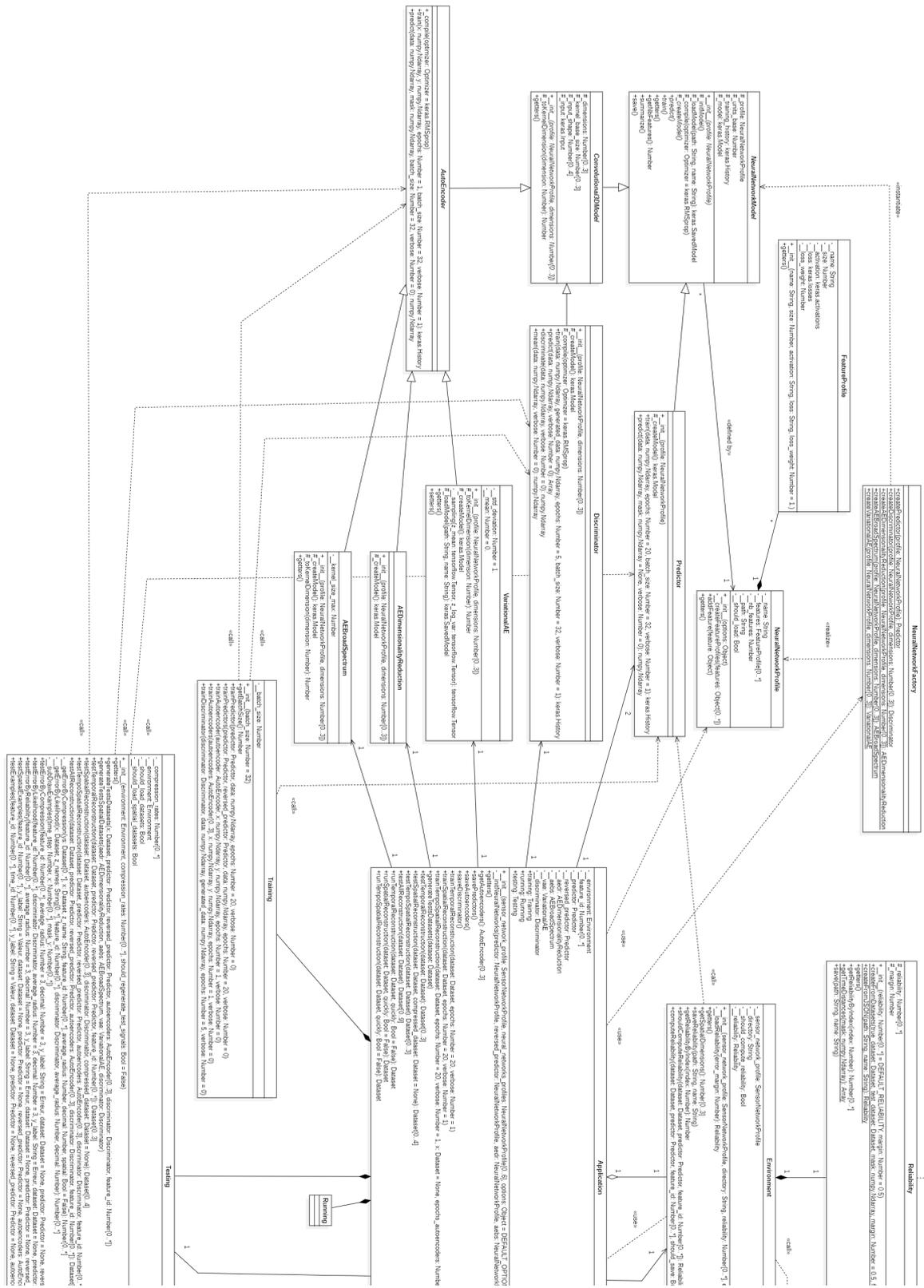
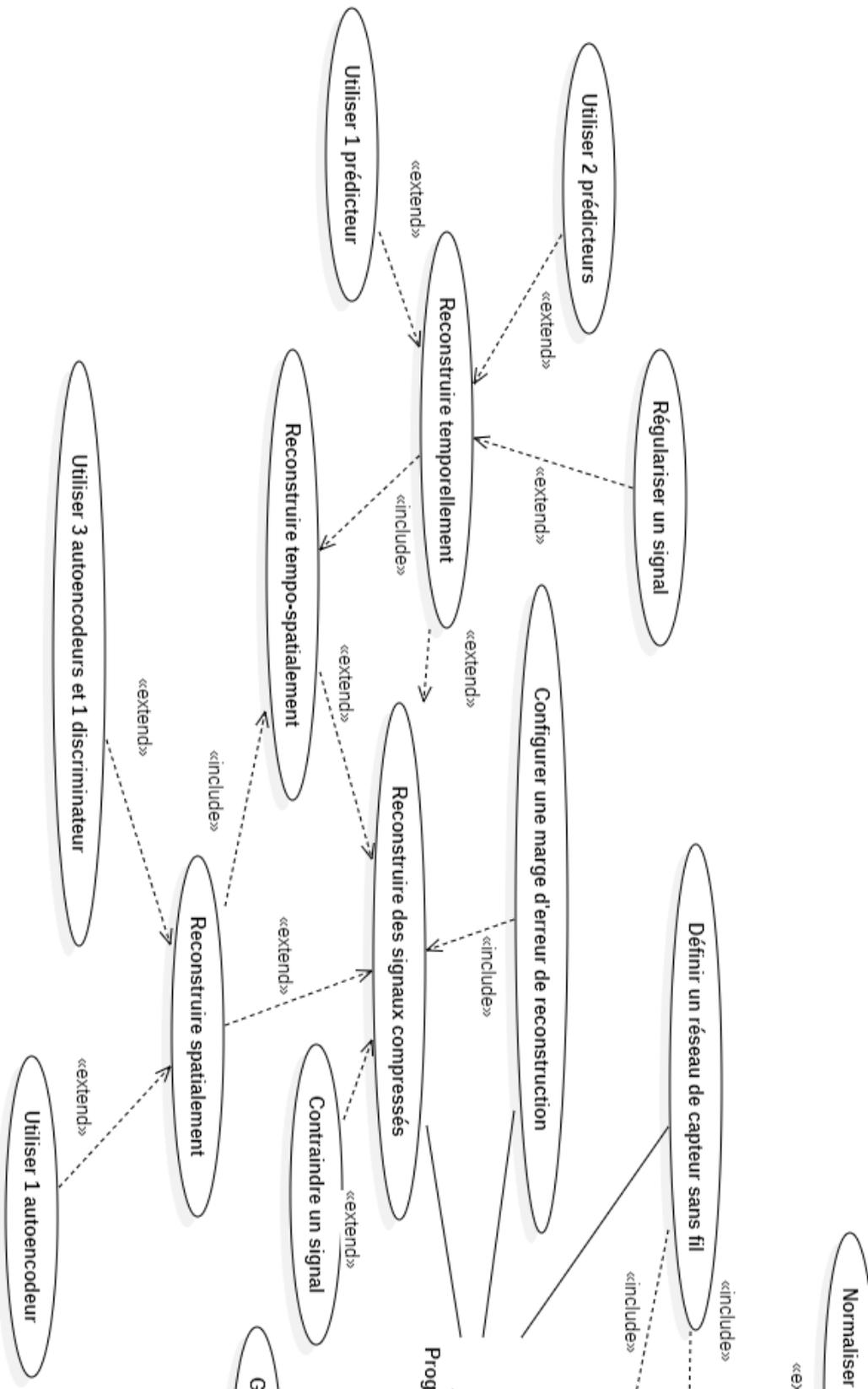


Figure A.1: Diagramme UML de paquetages de l'IPA deepsignals

A.1.2 Diagramme de classes



A.1.3 Diagramme de cas d'utilisation



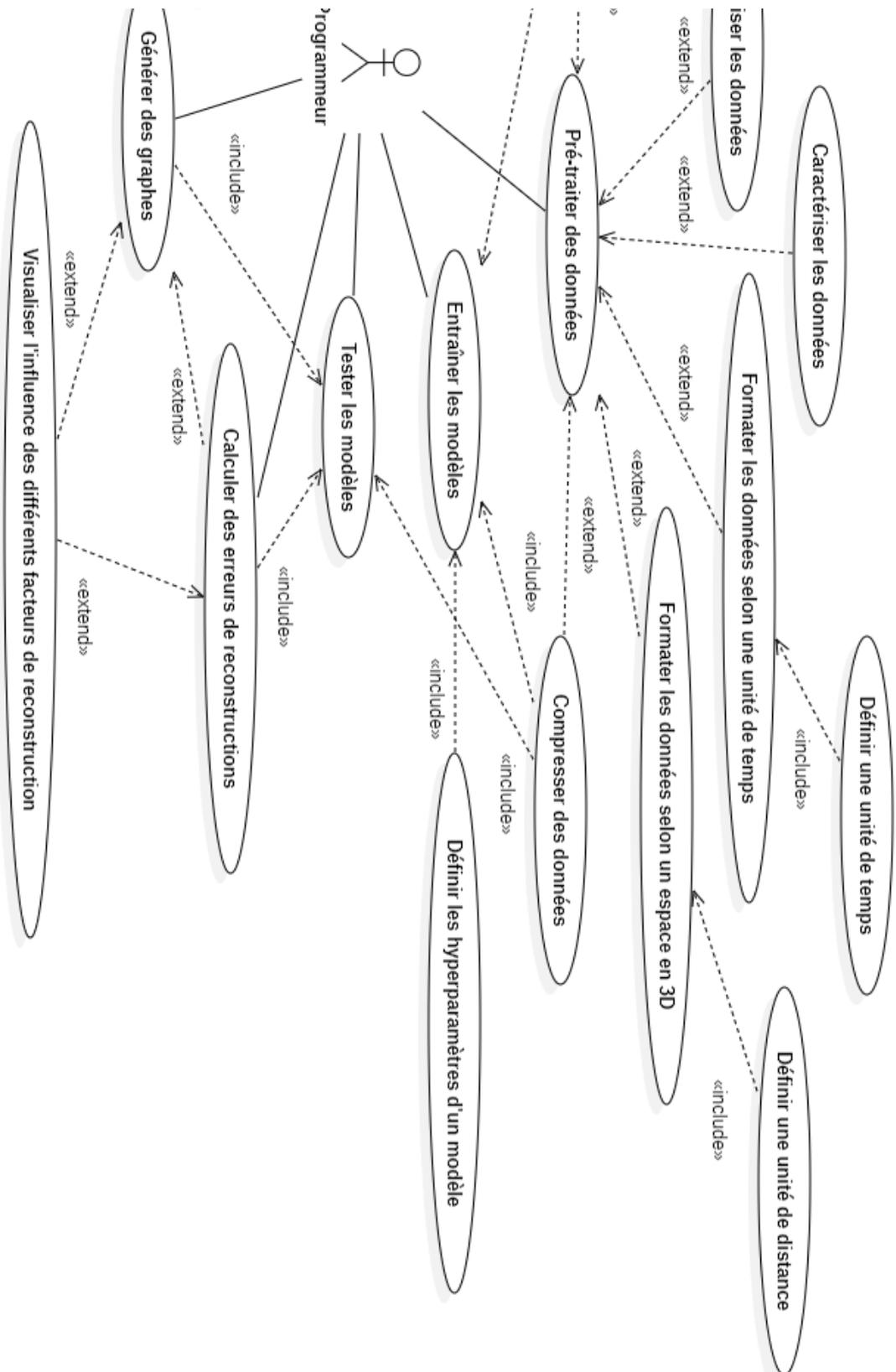


Figure A.3: Diagramme UML de cas d'utilisation de l'IPA deepsignals

A.2 Jeu de données

A.2.1 Signaux d'un réseau de capteur sans fil

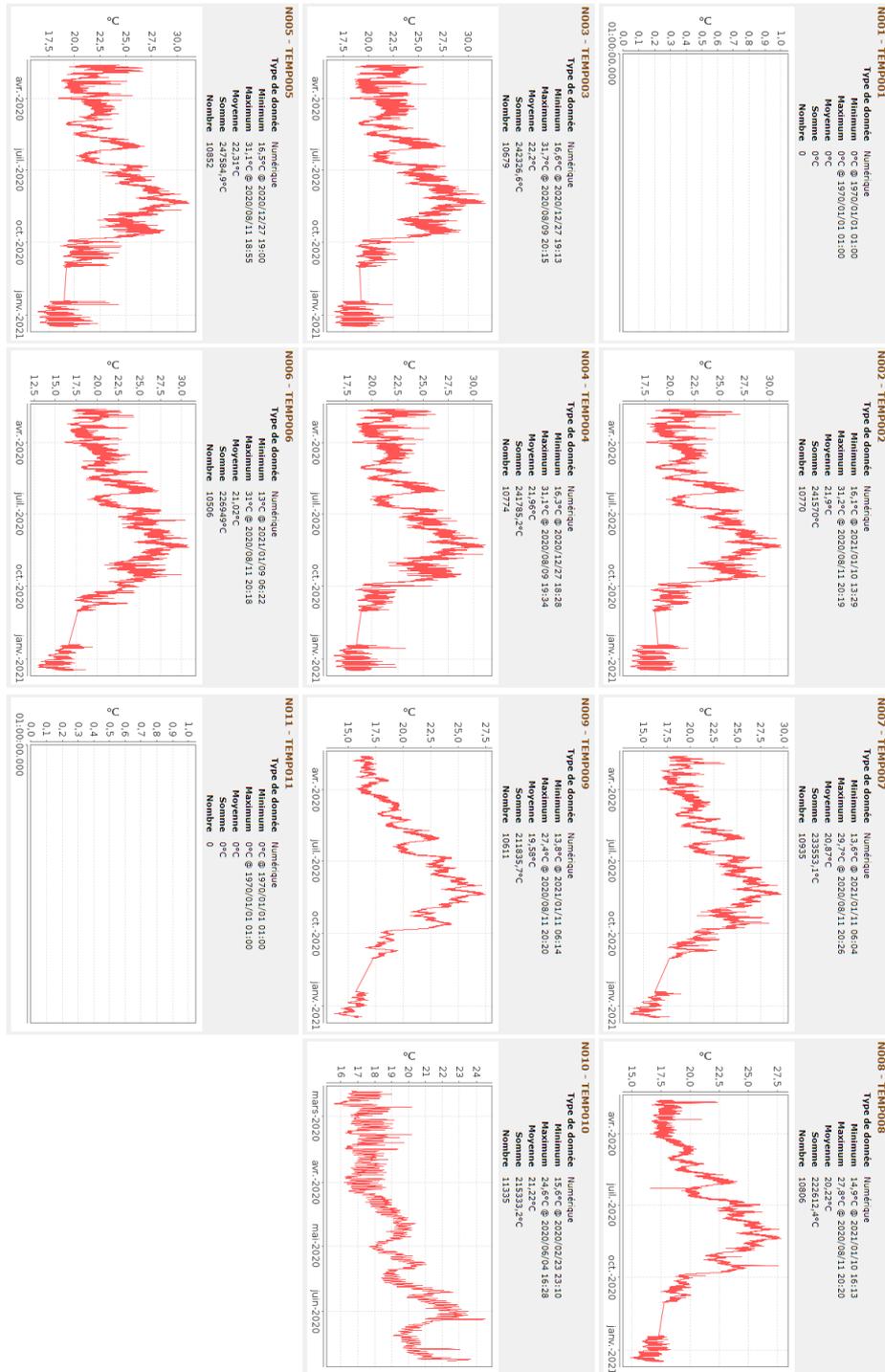


Figure A.4: Signaux de températures des noeuds de capteur du rez-de-chaussée du bâtiment GEII de Brive-la-Gaillarde

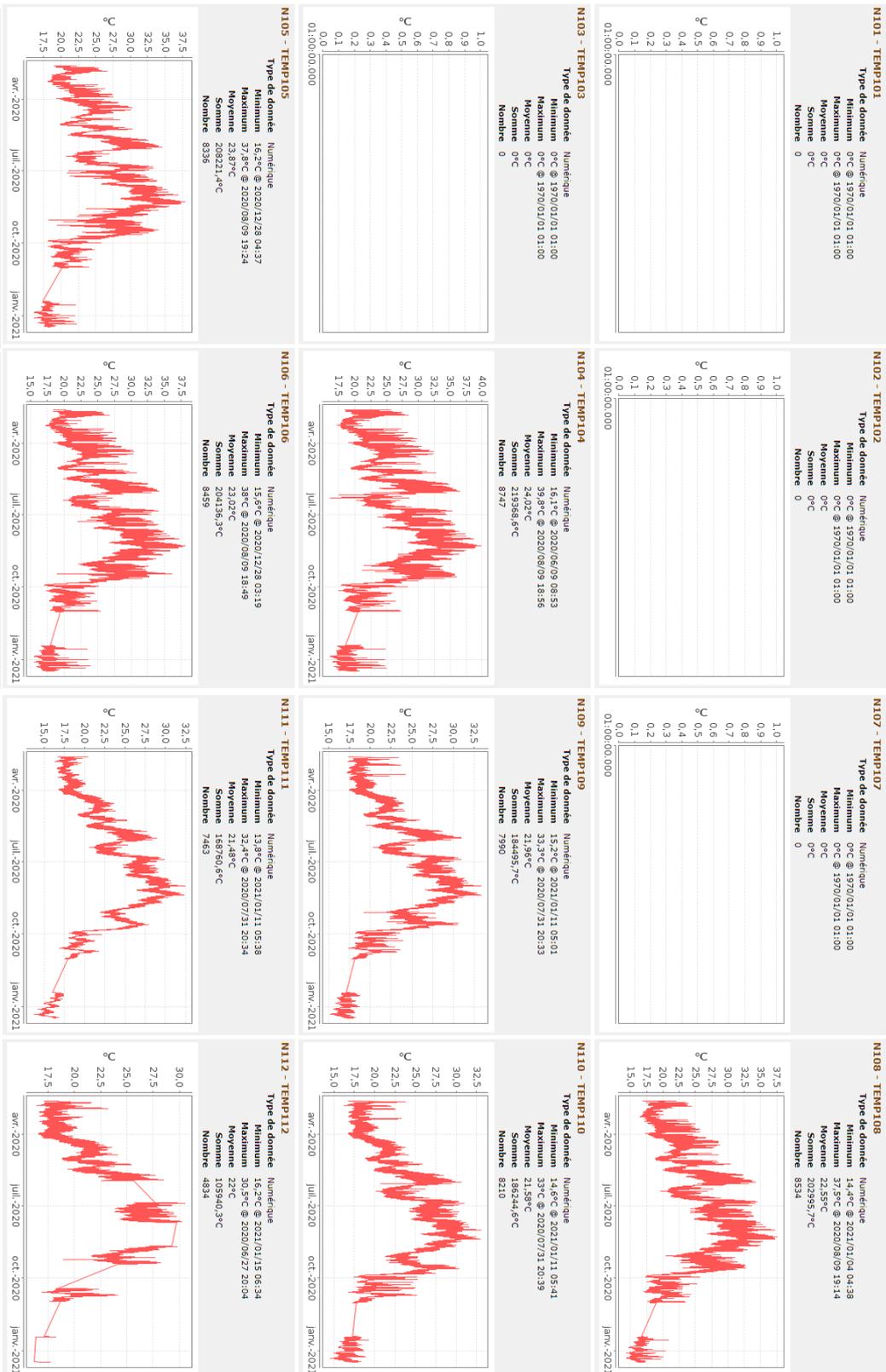
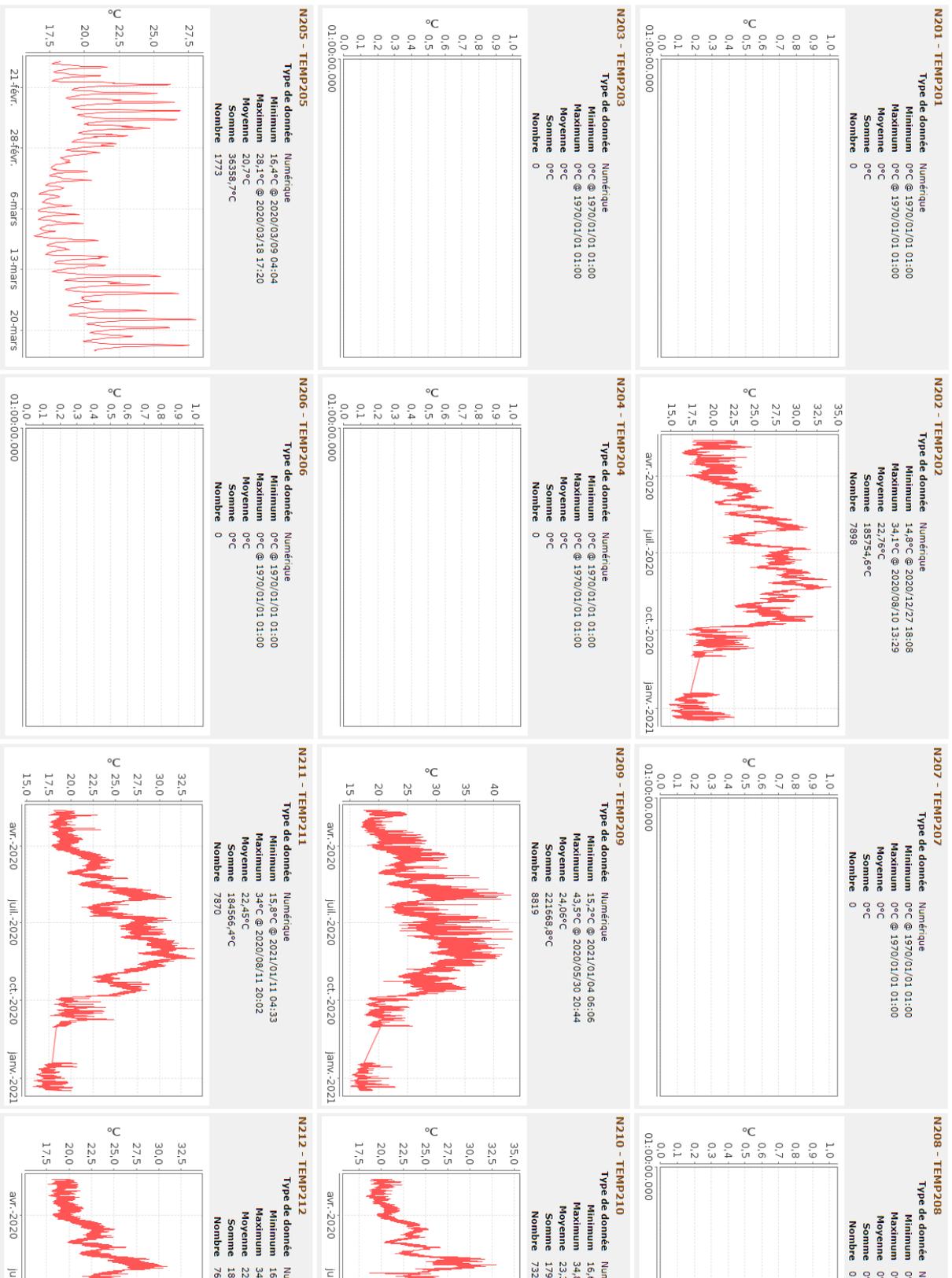


Figure A.5: Signaux de températures des noeuds de capteur du 1er étage du bâtiment GEII de Brive-la-Gaillarde



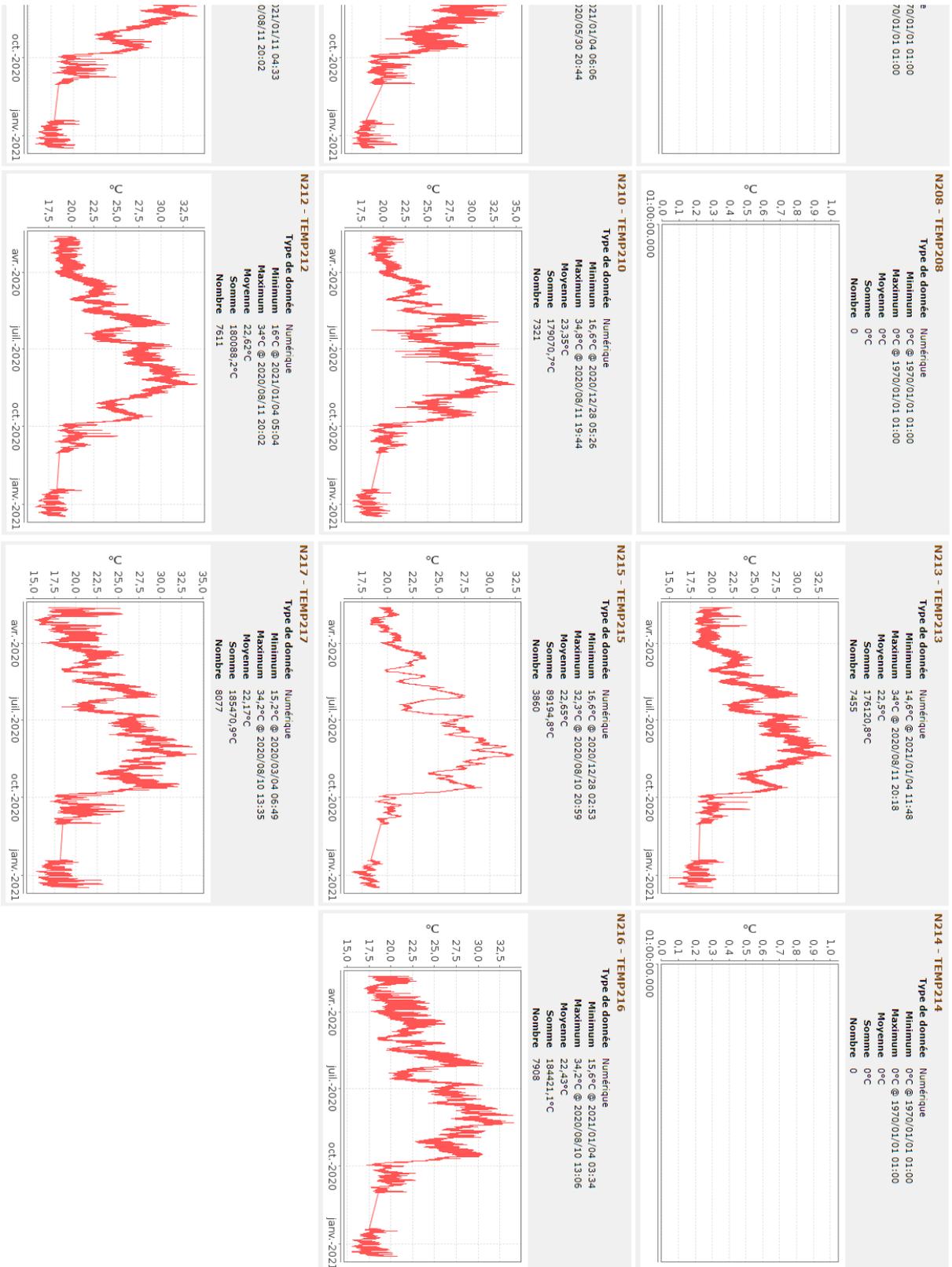


Figure A.6: Signaux de températures des noeuds de capteur du 2ème étage du bâtiment GEIL de Brive-la-Gaillarde

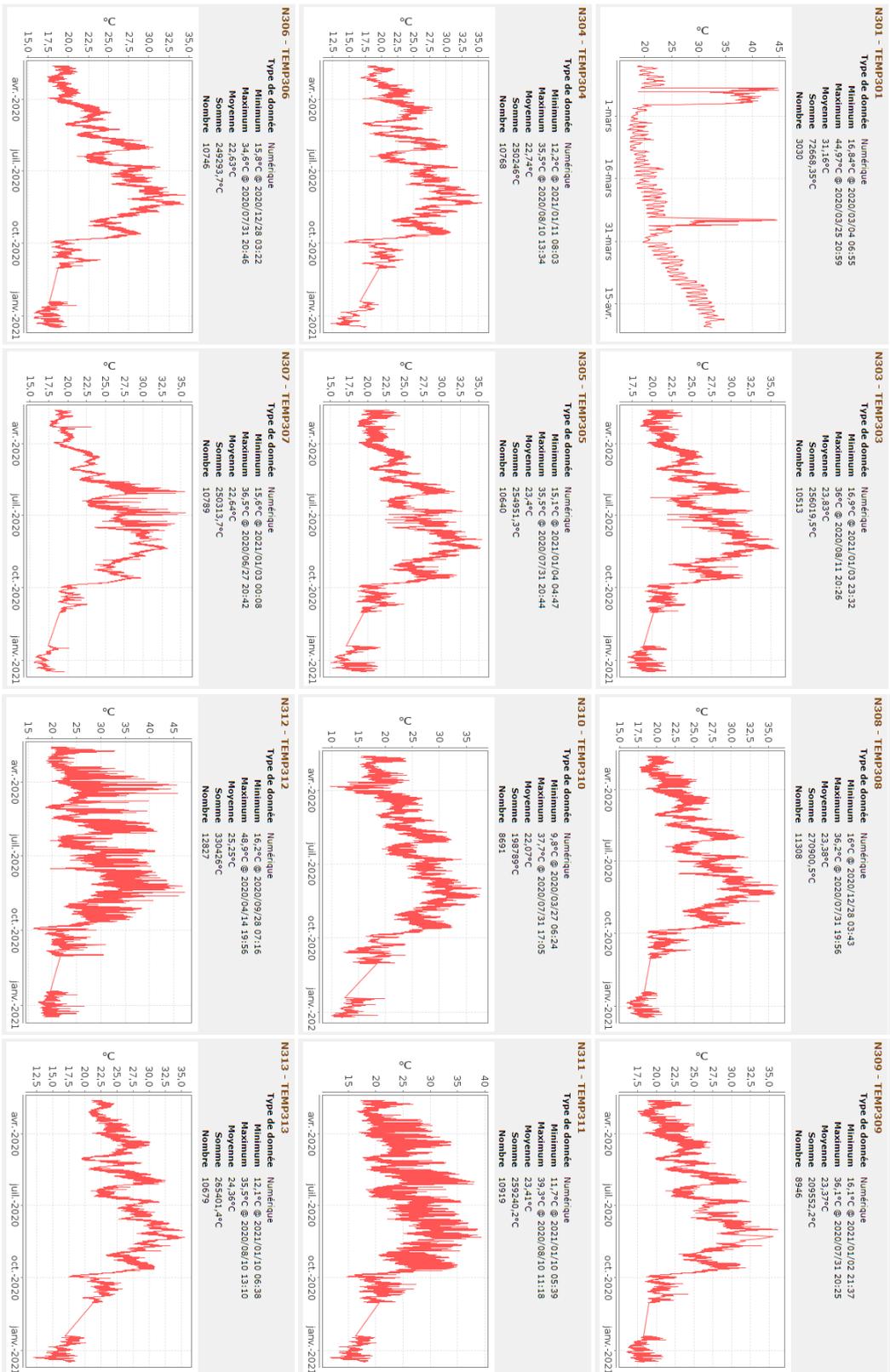


Figure A.7: Signaux de températures des noeuds de capteur du 3ème étage du bâtiment GEII de Brive-la-Gaillarde

B

Bibliographie

Sommaire

Références	162
Liste des travaux	173

Références

- [1] V. Potdar, A. Sharif, and E. Chang, “Wireless sensor networks: a survey,” in *2009 International Conference on Advanced Information Networking and Applications Workshops*, Bradford, United Kingdom: IEEE, May 2009, pp. 636–641, ISBN: 978-1-4244-3999-7. DOI: [10.1109/WAINA.2009.192](https://doi.org/10.1109/WAINA.2009.192).
- [2] S. Sharma and A. Kaur, “Survey on wireless sensor network, its applications and issues,” *Journal of Physics: Conference Series*, vol. 1969, no. 1, p. 012 042, Jul. 1, 2021, ISSN: 1742-6588, 1742-6596. DOI: [10.1088/1742-6596/1969/1/012042](https://doi.org/10.1088/1742-6596/1969/1/012042).
- [3] X. Wang, S. Wang, X. Liang, *et al.*, “Deep reinforcement learning: a survey,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2022, ISSN: 2162-237X, 2162-2388. DOI: [10.1109/TNNLS.2022.3207346](https://doi.org/10.1109/TNNLS.2022.3207346).
- [4] D. Donoho, “Compressed sensing,” *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006, ISSN: 0018-9448. DOI: [10.1109/TIT.2006.871582](https://doi.org/10.1109/TIT.2006.871582).
- [5] Y. C. Eldar and G. Kutyniok, Eds., *Compressed Sensing: Theory and Applications*, 1st ed., Cambridge University Press, May 17, 2012, ISBN: 978-1-107-00558-7. DOI: [10.1017/CB09780511794308](https://doi.org/10.1017/CB09780511794308).
- [6] D. Baron, M. F. Duarte, M. B. Wakin, S. Sarvotham, and R. G. Baraniuk, *Distributed compressive sensing*, Jan. 21, 2009. arXiv: [0901.3403\[cs,math\]](https://arxiv.org/abs/0901.3403).
- [7] H. Djelouat, A. Amira, and F. Bensaali, “Compressive sensing-based IoT applications: a review,” *Journal of Sensor and Actuator Networks*, vol. 7, no. 4, p. 45, Oct. 22, 2018, ISSN: 2224-2708. DOI: [10.3390/jsan7040045](https://doi.org/10.3390/jsan7040045).
- [8] P. Wei and F. He, “The compressed sensing of wireless sensor networks based on internet of things,” *IEEE Sensors Journal*, vol. 21, no. 22, pp. 25 267–25 273, Nov. 15, 2021, ISSN: 1530-437X, 1558-1748, 2379-9153. DOI: [10.1109/JSEN.2021.3071151](https://doi.org/10.1109/JSEN.2021.3071151).

- [9] F. Itoua Engoti, “Réalisation d’une plate-forme pour l’optimisation de réseaux de capteurs sans fil appliqués au bâtiment intelligent,” Ph.D. dissertation, Université de Limoges, 2018.
- [10] M. F. Duarte and R. G. Baraniuk, “Kronecker compressive sensing,” *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 494–504, Feb. 2012, ISSN: 1057-7149, 1941-0042. DOI: [10.1109/TIP.2011.2165289](https://doi.org/10.1109/TIP.2011.2165289).
- [11] E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, “A review of sparse recovery algorithms,” *IEEE Access*, vol. 7, pp. 1300–1322, 2019, ISSN: 2169-3536. DOI: [10.1109/ACCESS.2018.2886471](https://doi.org/10.1109/ACCESS.2018.2886471).
- [12] X. Lu, W. Dong, P. Wang, G. Shi, and X. Xie, *ConvCSNet: a convolutional compressive sensing framework based on deep learning*, Jan. 31, 2018. arXiv: [1801.10342](https://arxiv.org/abs/1801.10342)[cs].
- [13] A. Mousavi, G. Dasarathy, and R. G. Baraniuk, “A data-driven and distributed approach to sparse signal representation and recovery,” *ICLR 2019*, p. 13, 2018.
- [14] K. Xu and F. Ren, “CSVideoNet: a real-time end-to-end learning framework for high-frame-rate video compressive sensing,” in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV: IEEE, Mar. 2018, pp. 1680–1688, ISBN: 978-1-5386-4886-5. DOI: [10.1109/WACV.2018.00187](https://doi.org/10.1109/WACV.2018.00187).
- [15] Y. Yang, J. Sun, H. Li, and Z. Xu, “ADMM-CSNet: a deep learning approach for image compressive sensing,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 3, pp. 521–538, Mar. 1, 2020, ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: [10.1109/TPAMI.2018.2883941](https://doi.org/10.1109/TPAMI.2018.2883941).
- [16] Z. Zhang, Y. Liu, J. Liu, F. Wen, and C. Zhu, “AMP-net: denoising based deep unfolding for compressive image sensing,” *IEEE Transactions on Image Processing*, vol. 30, pp. 1487–1500, 2021, ISSN: 1057-7149, 1941-0042. DOI: [10.1109/TIP.2020.3044472](https://doi.org/10.1109/TIP.2020.3044472). arXiv: [2004.10078](https://arxiv.org/abs/2004.10078)[cs, eess].
- [17] E. C. Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, “Deep learning approaches for sparse recovery in compressive sensing,” in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, Dubrovnik, Croatia: IEEE, Sep. 2019, pp. 129–134, ISBN: 978-1-72813-140-5. DOI: [10.1109/ISPA.2019.8868841](https://doi.org/10.1109/ISPA.2019.8868841).
- [18] Z. Qin, J. Fan, Y. Liu, Y. Gao, and G. Y. Li, “Sparse representation for wireless communications: a compressive sensing approach,” *IEEE Signal Processing Magazine*, vol. 35, no. 3, pp. 40–58, May 2018, ISSN: 1053-5888. DOI: [10.1109/MSP.2018.2789521](https://doi.org/10.1109/MSP.2018.2789521).

- [19] A. L. Machidon and V. Pejovic, *Deep learning techniques for compressive sensing-based reconstruction and inference – a ubiquitous systems perspective*, May 26, 2021. arXiv: [2105.13191](https://arxiv.org/abs/2105.13191)[cs, eess].
- [20] H. Palangi, R. Ward, and L. Deng, “Distributed compressive sensing: a deep learning approach,” *IEEE Transactions on Signal Processing*, vol. 64, no. 17, pp. 4504–4518, Sep. 1, 2016, ISSN: 1053-587X, 1941-0476. DOI: [10.1109/TSP.2016.2557301](https://doi.org/10.1109/TSP.2016.2557301). arXiv: [1508.04924](https://arxiv.org/abs/1508.04924)[cs].
- [21] H. Palangi, R. Ward, and L. Deng, “Reconstruction of sparse vectors in compressive sensing with multiple measurement vectors using bidirectional long short-term memory,” in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Washington DC, DC, USA: IEEE, Dec. 2016, pp. 192–196, ISBN: 978-1-5090-4545-7. DOI: [10.1109/GlobalSIP.2016.7905830](https://doi.org/10.1109/GlobalSIP.2016.7905830).
- [22] C. Zhao, J. Liang, and Q. Tang, “A deep-learning-based distributed compressive sensing in UWB soil signals,” in *Communications, Signal Processing, and Systems*, Q. Liang, W. Wang, X. Liu, Z. Na, M. Jia, and B. Zhang, Eds., vol. 571, Series Title: Lecture Notes in Electrical Engineering, Singapore: Springer Singapore, 2020, pp. 1872–1879, ISBN: 9789811394089. DOI: [10.1007/978-981-13-9409-6_226](https://doi.org/10.1007/978-981-13-9409-6_226).
- [23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986, ISSN: 0028-0836, 1476-4687. DOI: [10.1038/323533a0](https://doi.org/10.1038/323533a0).
- [24] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, Dec. 1943, ISSN: 0007-4985, 1522-9602. DOI: [10.1007/BF02478259](https://doi.org/10.1007/BF02478259).
- [25] F. Rosenblatt, “Perceptron simulation experiments,” *Proceedings of the IRE*, vol. 48, no. 3, pp. 301–309, Mar. 1960, ISSN: 0096-8390. DOI: [10.1109/JRPROC.1960.287598](https://doi.org/10.1109/JRPROC.1960.287598).
- [26] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 28, 2015, ISSN: 0028-0836, 1476-4687. DOI: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [27] Y. LeCun, B. Boser, J. S. Denker, *et al.*, “Backpropagation applied to handwritten zip code recognition,” *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec. 1989, ISSN: 0899-7667, 1530-888X. DOI: [10.1162/neco.1989.1.4.541](https://doi.org/10.1162/neco.1989.1.4.541).
- [28] M. M. Bejani and M. Ghatee, “A systematic review on overfitting control in shallow and deep neural networks,” *Artificial Intelligence Review*, vol. 54, no. 8, pp. 6391–

- 6438, Dec. 2021, ISSN: 0269-2821, 1573-7462. DOI: [10.1007/s10462-021-09975-1](https://doi.org/10.1007/s10462-021-09975-1).
- [29] B. Ghojogh and M. Crowley, *The theory behind overfitting, cross validation, regularization, bagging, and boosting: tutorial*, May 20, 2023. arXiv: [1905.12787](https://arxiv.org/abs/1905.12787) [cs, stat].
- [30] S. Wang, J. Cao, and P. S. Yu, “Deep learning for spatio-temporal data mining: a survey,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3681–3700, Aug. 1, 2022, ISSN: 1041-4347, 1558-2191, 2326-3865. DOI: [10.1109/TKDE.2020.3025580](https://doi.org/10.1109/TKDE.2020.3025580).
- [31] S. Ruder, *An overview of gradient descent optimization algorithms*, Jun. 15, 2017. arXiv: [1609.04747](https://arxiv.org/abs/1609.04747) [cs].
- [32] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1, 1997, ISSN: 0899-7667, 1530-888X. DOI: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [33] D. Bahdanau, K. Cho, and Y. Bengio, *Neural machine translation by jointly learning to align and translate*, May 19, 2016. arXiv: [1409.0473](https://arxiv.org/abs/1409.0473) [cs, stat].
- [34] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, *Empirical evaluation of gated recurrent neural networks on sequence modeling*, Dec. 11, 2014. arXiv: [1412.3555](https://arxiv.org/abs/1412.3555) [cs].
- [35] M. Gori, G. Monfardini, and F. Scarselli, “A new model for learning in graph domains,” in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 2, Montreal, Que., Canada: IEEE, 2005, pp. 729–734, ISBN: 978-0-7803-9048-5. DOI: [10.1109/IJCNN.2005.1555942](https://doi.org/10.1109/IJCNN.2005.1555942).
- [36] F. Scarselli, M. Gori, Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, Jan. 2009, ISSN: 1045-9227, 1941-0093. DOI: [10.1109/TNN.2008.2005605](https://doi.org/10.1109/TNN.2008.2005605).
- [37] F. Liang, C. Qian, W. Yu, D. Griffith, and N. Golmie, “Survey of graph neural networks and applications,” *Wireless Communications and Mobile Computing*, vol. 2022, W. Li, Ed., pp. 1–18, Jul. 28, 2022, ISSN: 1530-8677, 1530-8669. DOI: [10.1155/2022/9261537](https://doi.org/10.1155/2022/9261537).
- [38] K. Fukushima, “Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, Apr. 1980, ISSN: 0340-1200, 1432-0770. DOI: [10.1007/BF00344251](https://doi.org/10.1007/BF00344251).

- [39] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, “A survey of convolutional neural networks: analysis, applications, and prospects,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, ISSN: 2162-237X, 2162-2388. DOI: [10.1109/TNNLS.2021.3084827](https://doi.org/10.1109/TNNLS.2021.3084827).
- [40] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo, “Convolutional LSTM network: a machine learning approach for precipitation nowcasting,” *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, 2015.
- [41] T. N. Kipf and M. Welling, *Semi-supervised classification with graph convolutional networks*, Feb. 22, 2017. arXiv: [1609.02907](https://arxiv.org/abs/1609.02907) [cs, stat].
- [42] L. Alzubaidi, J. Zhang, A. J. Humaidi, *et al.*, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” *Journal of Big Data*, vol. 8, no. 1, p. 53, Dec. 2021, ISSN: 2196-1115. DOI: [10.1186/s40537-021-00444-8](https://doi.org/10.1186/s40537-021-00444-8).
- [43] N. Shawki, R. R. Nunez, I. Obeid, and J. Picone, “On automating hyperparameter optimization for deep learning applications,” in *2021 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, Philadelphia, PA, USA: IEEE, Dec. 4, 2021, pp. 1–7, ISBN: 978-1-66542-897-2. DOI: [10.1109/SPMB52430.2021.9672266](https://doi.org/10.1109/SPMB52430.2021.9672266).
- [44] P. Li, Y. Pei, and J. Li, “A comprehensive survey on design and application of autoencoder in deep learning,” *Applied Soft Computing*, vol. 138, p. 110 176, May 2023, ISSN: 15684946. DOI: [10.1016/j.asoc.2023.110176](https://doi.org/10.1016/j.asoc.2023.110176).
- [45] P. Pradhyumna and Mohana, “A survey of modern deep learning based generative adversarial networks (GANs),” in *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India: IEEE, Mar. 29, 2022, pp. 1146–1152, ISBN: 978-1-66541-028-1. DOI: [10.1109/ICCMC53470.2022.9753782](https://doi.org/10.1109/ICCMC53470.2022.9753782).
- [46] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein GAN*, Dec. 6, 2017. arXiv: [1701.07875](https://arxiv.org/abs/1701.07875) [cs, stat].
- [47] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, Dec. 5, 2017. arXiv: [1706.03762](https://arxiv.org/abs/1706.03762) [cs].
- [48] T. Lin, Y. Wang, X. Liu, and X. Qiu, “A survey of transformers,” *AI Open*, vol. 3, pp. 111–132, 2022, ISSN: 26666510. DOI: [10.1016/j.aiopen.2022.10.001](https://doi.org/10.1016/j.aiopen.2022.10.001).
- [49] S. Islam, H. Elmekki, A. Elsebai, *et al.*, *A comprehensive survey on applications of transformers for deep learning tasks*, Jun. 11, 2023. arXiv: [2306.07303](https://arxiv.org/abs/2306.07303) [cs].
- [50] L. Yang, Z. Zhang, Y. Song, *et al.*, *Diffusion models: a comprehensive survey of methods and applications*, Mar. 23, 2023. arXiv: [2209.00796](https://arxiv.org/abs/2209.00796) [cs].

- [51] K. Benidis, S. S. Rangapuram, V. Flunkert, *et al.*, “Deep learning for time series forecasting: tutorial and literature survey,” *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–36, Jul. 31, 2023, ISSN: 0360-0300, 1557-7341. DOI: [10.1145/3533382](https://doi.org/10.1145/3533382). arXiv: [2004.10240\[cs,stat\]](https://arxiv.org/abs/2004.10240).
- [52] H. Zhou, S. Zhang, J. Peng, *et al.*, *Informer: beyond efficient transformer for long sequence time-series forecasting*, Mar. 28, 2021. arXiv: [2012.07436\[cs\]](https://arxiv.org/abs/2012.07436).
- [53] H. Hewamalage, C. Bergmeir, and K. Bandara, “Recurrent neural networks for time series forecasting: current status and future directions,” *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, Jan. 2021, ISSN: 01692070. DOI: [10.1016/j.ijforecast.2020.06.008](https://doi.org/10.1016/j.ijforecast.2020.06.008).
- [54] M. Akram and C. El, “Sequence to sequence weather forecasting with long short-term memory recurrent neural networks,” *International Journal of Computer Applications*, vol. 143, no. 11, pp. 7–11, Jun. 17, 2016, ISSN: 09758887. DOI: [10.5120/ijca2016910497](https://doi.org/10.5120/ijca2016910497).
- [55] J. Duan, H. Zuo, Y. Bai, J. Duan, M. Chang, and B. Chen, “Short-term wind speed forecasting using recurrent neural networks with error correction,” *Energy*, vol. 217, p. 119397, Feb. 2021, ISSN: 03605442. DOI: [10.1016/j.energy.2020.119397](https://doi.org/10.1016/j.energy.2020.119397).
- [56] S. Farah, W. David A, N. Humaira, Z. Aneela, and E. Steffen, “Short-term multi-hour ahead country-wide wind power prediction for germany using gated recurrent unit deep learning,” *Renewable and Sustainable Energy Reviews*, vol. 167, p. 112700, Oct. 2022, ISSN: 13640321. DOI: [10.1016/j.rser.2022.112700](https://doi.org/10.1016/j.rser.2022.112700).
- [57] A. Kisvari, Z. Lin, and X. Liu, “Wind power forecasting – a data-driven method along with gated recurrent neural network,” *Renewable Energy*, vol. 163, pp. 1895–1909, Jan. 2021, ISSN: 09601481. DOI: [10.1016/j.renene.2020.10.119](https://doi.org/10.1016/j.renene.2020.10.119).
- [58] X. Tang, Y. Dai, T. Wang, and Y. Chen, “Short-term power load forecasting based on multi-layer bidirectional recurrent neural network,” *IET Generation, Transmission & Distribution*, vol. 13, no. 17, pp. 3847–3854, Sep. 2019, ISSN: 1751-8695, 1751-8695. DOI: [10.1049/iet-gtd.2018.6687](https://doi.org/10.1049/iet-gtd.2018.6687).
- [59] J. Wang, Q. Li, H. Zhang, and Y. Wang, “A deep-learning wind speed interval forecasting architecture based on modified scaling approach with feature ranking and two-output gated recurrent unit,” *Expert Systems with Applications*, vol. 211, p. 118419, Jan. 2023, ISSN: 09574174. DOI: [10.1016/j.eswa.2022.118419](https://doi.org/10.1016/j.eswa.2022.118419).

- [60] K. H. Waseem, H. Mushtaq, F. Abid, *et al.*, “Forecasting of air quality using an optimized recurrent neural network,” *Processes*, vol. 10, no. 10, p. 2117, Oct. 18, 2022, ISSN: 2227-9717. DOI: [10.3390/pr10102117](https://doi.org/10.3390/pr10102117).
- [61] A. Ghaderi, B. M. Sanandaji, and F. Ghaderi, *Deep forecast: deep learning-based spatio-temporal forecasting*, Jul. 24, 2017. arXiv: [1707.08110](https://arxiv.org/abs/1707.08110) [cs].
- [62] Z. Zhang and Y. Dong, “Temperature forecasting via convolutional recurrent neural networks based on time-series data,” *Complexity*, vol. 2020, pp. 1–8, Mar. 20, 2020, ISSN: 1076-2787, 1099-0526. DOI: [10.1155/2020/3536572](https://doi.org/10.1155/2020/3536572).
- [63] P. Newbold, “ARIMA model building and the time series analysis approach to forecasting,” *Journal of Forecasting*, vol. 2, no. 1, pp. 23–35, Jan. 1983, ISSN: 02776693, 1099131X. DOI: [10.1002/for.3980020104](https://doi.org/10.1002/for.3980020104).
- [64] E. De Saa and L. Ranathunga, “Comparison between ARIMA and deep learning models for temperature forecasting,” 2020, Publisher: arXiv Version Number: 1. DOI: [10.48550/ARXIV.2011.04452](https://doi.org/10.48550/ARXIV.2011.04452).
- [65] V. I. Kontopoulou, A. D. Panagopoulos, I. Kakkos, and G. K. Matsopoulos, “A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks,” *Future Internet*, vol. 15, no. 8, p. 255, Jul. 30, 2023, ISSN: 1999-5903. DOI: [10.3390/fi15080255](https://doi.org/10.3390/fi15080255).
- [66] J. R. M. Hosking, “Fractional differencing,” *Biometrika*, vol. 68, no. 1, pp. 165–176, 1981, ISSN: 0006-3444, 1464-3510. DOI: [10.1093/biomet/68.1.165](https://doi.org/10.1093/biomet/68.1.165).
- [67] R. Zhang, H. Song, Q. Chen, Y. Wang, S. Wang, and Y. Li, “Comparison of ARIMA and LSTM for prediction of hemorrhagic fever at different time scales in china,” *PLOS ONE*, vol. 17, no. 1, E. Tlelo-Cuautle, Ed., e0262009, Jan. 14, 2022, ISSN: 1932-6203. DOI: [10.1371/journal.pone.0262009](https://doi.org/10.1371/journal.pone.0262009).
- [68] Y. Balagula and I. Aharon, *A comparison of statistical and machine learning models for long memory in electricity price forecasting*, 2024. DOI: [10.2139/ssrn.4803433](https://doi.org/10.2139/ssrn.4803433).
- [69] M. Delavari, N. Gandali Alikhani, and E. Naderi, “Do dynamic neural networks stand a better chance in fractionally integrated process forecasting?” *International Journal of Economics and Financial Issues*, pp. 466–475, Apr. 2013.
- [70] S. Khedhiri, “Comparison of SARFIMA and LSTM methods to model and to forecast canadian temperature,” *Regional Statistics*, vol. 12, no. 2, pp. 177–194, 2022, ISSN: 20639538, 20648243. DOI: [10.15196/RS120204](https://doi.org/10.15196/RS120204).
- [71] F. Saâdaoui and H. Rabbouch, “Financial forecasting improvement with LSTM-ARFIMA hybrid models and non-gaussian distributions,” *Technological Forecast-*

- ing and Social Change*, vol. 206, p. 123 539, Sep. 2024, ISSN: 00401625. DOI: [10.1016/j.techfore.2024.123539](https://doi.org/10.1016/j.techfore.2024.123539).
- [72] Y. Wang, M. Long, J. Wang, Z. Gao, and P. S. Yu, “PredRNN: recurrent neural networks for predictive learning using spatiotemporal LSTMs,” *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, Dec. 4, 2017.
- [73] Y. Wang, Z. Gao, M. Long, J. Wang, and P. S. Yu, *PredRNN++: towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning*, Nov. 18, 2018. arXiv: [1804.06300](https://arxiv.org/abs/1804.06300)[cs, stat].
- [74] Y. Wang, H. Wu, J. Zhang, *et al.*, “PredRNN: a recurrent neural network for spatiotemporal predictive learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 2208–2225, Feb. 1, 2023, ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: [10.1109/TPAMI.2022.3165153](https://doi.org/10.1109/TPAMI.2022.3165153).
- [75] B. Nikparvar and J.-C. Thill, “Machine learning of spatial data,” *ISPRS International Journal of Geo-Information*, vol. 10, no. 9, p. 600, Sep. 12, 2021, ISSN: 2220-9964. DOI: [10.3390/ijgi10090600](https://doi.org/10.3390/ijgi10090600).
- [76] Q. Zhu, J. Chen, L. Zhu, X. Duan, and Y. Liu, “Wind speed prediction with spatio-temporal correlation: a deep learning approach,” *Energies*, vol. 11, no. 4, p. 705, Mar. 21, 2018, ISSN: 1996-1073. DOI: [10.3390/en11040705](https://doi.org/10.3390/en11040705).
- [77] S. Giffard-Roisin, M. Yang, G. Charpiat, C. Kumler-Bonfanti, B. Kégl, and C. Monteleoni, “Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data,” *Frontiers in Big Data*, vol. 3, p. 1, Jan. 28, 2020, ISSN: 2624-909X. DOI: [10.3389/fdata.2020.00001](https://doi.org/10.3389/fdata.2020.00001). arXiv: [1910.10566](https://arxiv.org/abs/1910.10566)[physics, stat].
- [78] T. Wang, Y. Zhang, X. Zhi, and Y. Ji, “Multi-model ensemble forecasts of surface air temperatures in henan province based on machine learning,” *Atmosphere*, vol. 14, no. 3, p. 520, Mar. 8, 2023, ISSN: 2073-4433. DOI: [10.3390/atmos14030520](https://doi.org/10.3390/atmos14030520).
- [79] F. Amato, F. Guignard, S. Robert, and M. Kanevski, “A novel framework for spatio-temporal prediction of environmental data using deep learning,” *Scientific Reports*, vol. 10, no. 1, p. 22 243, Dec. 17, 2020, ISSN: 2045-2322. DOI: [10.1038/s41598-020-79148-7](https://doi.org/10.1038/s41598-020-79148-7). arXiv: [2007.11836](https://arxiv.org/abs/2007.11836)[physics, stat].
- [80] L. Ø. Bentsen, N. D. Warakagoda, R. Stenbro, and P. Engelstad, “Spatio-temporal wind speed forecasting using graph networks and novel transformer architectures,” *Applied Energy*, vol. 333, p. 120 565, Mar. 2023, ISSN: 03062619. DOI: [10.1016/j.apenergy.2022.120565](https://doi.org/10.1016/j.apenergy.2022.120565).

- [81] R. Sen, H.-F. Yu, and I. Dhillon, *Think globally, act locally: a deep neural network approach to high-dimensional time series forecasting*, Oct. 26, 2019. arXiv: [1905.03806\[cs,stat\]](#).
- [82] N. B. Ipsen, P.-A. Mattei, and J. Frellsen, “How to deal with missing data in supervised deep learning?” *ICLR 2022*, Jan. 28, 2022.
- [83] Y. Luo, Y. Zhang, X. Cai, and X. Yuan, “E²GAN: end-to-end generative adversarial network for multivariate time series imputation,” in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, Macao, China: International Joint Conferences on Artificial Intelligence Organization, Aug. 2019, pp. 3094–3100, ISBN: 978-0-9992411-4-1. DOI: [10.24963/ijcai.2019/429](#).
- [84] Y. Zhang, B. Zhou, X. Cai, W. Guo, X. Ding, and X. Yuan, “Missing value imputation in multivariate time series with end-to-end generative adversarial networks,” *Information Sciences*, vol. 551, pp. 67–82, Apr. 2021, ISSN: 00200255. DOI: [10.1016/j.ins.2020.11.035](#).
- [85] X. Miao, Y. Wu, J. Wang, Y. Gao, X. Mao, and J. Yin, “Generative semi-supervised learning for multivariate time series imputation,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 10, pp. 8983–8991, May 18, 2021, ISSN: 2374-3468, 2159-5399. DOI: [10.1609/aaai.v35i10.17086](#).
- [86] S. Yang, M. Dong, Y. Wang, and C. Xu, “Adversarial recurrent time series imputation,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 4, pp. 1639–1650, Apr. 2023, ISSN: 2162-237X, 2162-2388. DOI: [10.1109/TNNLS.2020.3010524](#).
- [87] R. Qin and Y. Wang, “ImputeGAN: generative adversarial network for multivariate time series imputation,” *Entropy*, vol. 25, no. 1, p. 137, Jan. 10, 2023, ISSN: 1099-4300. DOI: [10.3390/e25010137](#).
- [88] N. Gao, H. Xue, W. Shao, *et al.*, “Generative adversarial networks for spatio-temporal data: a survey,” *ACM Transactions on Intelligent Systems and Technology*, vol. 13, no. 2, pp. 1–25, Apr. 30, 2022, ISSN: 2157-6904, 2157-6912. DOI: [10.1145/3474838](#). arXiv: [2008.08903\[cs,eess\]](#).
- [89] C. Fang and C. Wang, *Time series data imputation: a survey on deep learning approaches*, Nov. 23, 2020. arXiv: [2011.11347\[cs\]](#).
- [90] Y. Tashiro, J. Song, Y. Song, and S. Ermon, *CSDI: conditional score-based diffusion models for probabilistic time series imputation*, Oct. 27, 2021. arXiv: [2107.03502\[cs,stat\]](#).

- [91] J. M. L. Alcaraz and N. Strodthoff, *Diffusion-based time series imputation and forecasting with structured state space models*, May 6, 2023. arXiv: [2208.09399\[cs,stat\]](#).
- [92] W. Yu, C. Mechefske, and I. Y. Kim, "Time series reconstruction using a bidirectional recurrent neural network based encoder-decoder scheme," *HUMS 2019*, May 2019.
- [93] K. Aggarwal and J. Srivastava, *Filling out the missing gaps: time series imputation with semi-supervised learning*, Apr. 9, 2023. arXiv: [2304.04275\[cs\]](#).
- [94] W. Cao, D. Wang, J. Li, H. Zhou, L. Li, and Y. Li, "BRITS : bidirectional recurrent imputation for time series," *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, 2018.
- [95] M. Liu, H. Huang, H. Feng, L. Sun, B. Du, and Y. Fu, *PriSTI: a conditional diffusion framework for spatiotemporal imputation*, Feb. 19, 2023. arXiv: [2302.09746\[cs\]](#).
- [96] A. Cini, I. Marisca, and C. Alippi, "Filling the g_ap_s: multivariate time series imputation by graph neural networks," *ICLR 2022*, 2022.
- [97] I. Kok and S. Ozdemir, "DeepMDP: a novel deep-learning-based missing data prediction protocol for IoT," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 232–243, Jan. 1, 2021, ISSN: 2327-4662, 2372-2541. DOI: [10.1109/JIOT.2020.3003922](#).
- [98] S. Badri, M. Saini, and N. Goel, *Design of energy harvesting based hardware for IoT applications*, Jun. 21, 2023. arXiv: [2306.12019\[cs\]](#).
- [99] Y. Zhou, S. Wang, T. Wu, *et al.*, "For-backward LSTM-based missing data reconstruction for time-series landsat images," *GIScience & Remote Sensing*, vol. 59, no. 1, pp. 410–430, Dec. 31, 2022, ISSN: 1548-1603, 1943-7226. DOI: [10.1080/15481603.2022.2031549](#).
- [100] S. Ioffe and C. Szegedy, *Batch normalization: accelerating deep network training by reducing internal covariate shift*, Mar. 2, 2015. arXiv: [1502.03167\[cs\]](#).
- [101] M. Cerbelaud, K. Lebdioua, C. T. Tran, B. Crespín, A. Aimable, and A. Videcoq, "Brownian dynamics simulations of one-patch inverse patchy particles," *Physical Chemistry Chemical Physics*, vol. 21, no. 42, pp. 23 447–23 458, 2019, ISSN: 1463-9076, 1463-9084. DOI: [10.1039/C9CP04247D](#).
- [102] G. Rosenthal, K. E. Gubbins, and S. H. L. Klapp, "Self-assembly of model amphiphilic janus particles," *The Journal of Chemical Physics*, vol. 136, no. 17, p. 174 901, May 7, 2012, ISSN: 0021-9606, 1089-7690. DOI: [10.1063/1.4707954](#).
- [103] K. Lebdioua, M. Cerbelaud, A. Aimable, and A. Videcoq, "Study of the aggregation behavior of janus particles by coupling experiments and brownian dynamics

- simulations,” *Journal of Colloid and Interface Science*, vol. 583, pp. 222–233, Feb. 2021, ISSN: 00219797. DOI: [10.1016/j.jcis.2020.09.031](https://doi.org/10.1016/j.jcis.2020.09.031).
- [104] D. Kushnir and L. Demkiv, “Identification of dynamical system’s parameters using neural networks,” Ph.D. dissertation, Ukrainian Catholic University, Lviv, 2019, 61 pp.
- [105] R. A. DeLaCruz-Araujo, D. J. Beltran-Villegas, R. G. Larson, and U. M. Córdoba-Figueroa, “Rich janus colloid phase behavior under steady shear,” *Soft Matter*, vol. 12, no. 18, pp. 4071–4081, 2016, ISSN: 1744-683X, 1744-6848. DOI: [10.1039/C6SM00183A](https://doi.org/10.1039/C6SM00183A).

Liste des Travaux

Conférences internationales à comité de lecture

S. Rousseau, R. Négrier, F. Courreges, F. I. Engoti, and R. Quéré, “Higher reconstruction performance with dual deep neural networks for wireless sensor networks compressed temporal signals,” in *2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Dortmund, Germany: IEEE, Sep. 7, 2023, pp. 244–248, ISBN: 9798350358056. DOI: [10.1109/IDAACS58523.2023.10348666](https://doi.org/10.1109/IDAACS58523.2023.10348666).

Développement de méthodes d'apprentissage profond pour la reconstruction de données spatio-temporelles hautement comprimées – application aux signaux des réseaux de capteurs sans fil et aux simulations céramiques de structures colloïdales

Résumé : Cette thèse présente des méthodes d'apprentissage profond de reconstruction de données spatio-temporelles qui identifient les modèles de transformation des données connues en données attendues. Une première application est dédiée aux signaux compressés de réseaux de capteurs sans fil. Elle est basée sur la théorie de l'acquisition comprimée qui permet de reconstruire un signal à partir d'un signal compressé. Une seconde application est dédiée aux données de simulation de structures colloïdales de l'institut de recherche sur les céramiques IRCER. Elle est basée sur les théories de modèles d'interaction de particules qui estiment les données à partir de paramètres qu'il faut identifier. La thèse s'appuie sur une étude de l'état de l'art 2023 en matière d'apprentissage profond en vue de le surpasser pour les applications qui la concerne. Ce manuscrit détaille les méthodes et modèles depuis leur conception, au développement et à l'expérimentation en répondant aux problématiques des réseaux de capteurs sans fil et des simulations de structures colloïdales.

Mots clés : apprentissage profond, réseaux de neurones, acquisition comprimée, réseau de capteur sans fil, identification paramétrique, colloïdes

Deep learning methods development for highly compressed spatio-temporal data reconstruction - application on wireless sensor networks signals and colloids ceramic simulations

Abstract: This thesis introduces advanced deep learning techniques for the reconstruction of spatio-temporal data. The primary objective was to devise highly efficient models capable of accurately restoring missing data within multidimensional signal datasets with limited information. The first application leverages compressed sensing theory to reconstruct wireless sensor networks signals that are sparsely represented. The second application employs colloid simulation data from the IRCER ceramics research institute for the system identification of particle interaction models. The thesis builds upon a comprehensive 2023 state-of-the-art study on deep learning, aiming not only to incorporate its findings but also to surpass its achievements in the specific applications addressed. This manuscript provides a detailed account of the entire process, spanning from the conception of methods and models to their development and experimentation, addressing challenges in wireless sensor networks and colloid simulations.

Keywords: deep learning, neural networks, compressed sensing, wireless sensor network, system identification, colloids

Univ. de Limoges, CNRS, XLIM, UMR 7252, F-87000 Limoges, France