Pilotage distribué de systèmes multi-énergies en réseau

Pierre Blaud

▶ **To cite this version:**

Pierre Blaud. Pilotage distribué de systèmes multi-énergies en réseau. Automatique. Ecole nationale supérieure Mines-Télécom Atlantique, 2022. Français. NNT : 2022IMTA0290 . tel-03681888

HAL Id: tel-03681888
https://theses.hal.science/tel-03681888

Submitted on 30 May 2022

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.

DOCTORAT BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE
MINES-TÉLÉCOM ATLANTIQUE BRETAGNE
PAYS-DE-LA-LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Automatique*

Par
Pierre BLAUD

Pilotage distribué de systèmes multi-énergies en réseau

Thèse présentée et soutenue à IMT Atlantique, Nantes, le 21 Mars 2022
Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)
Thèse N° : 2022IMTA0290

Rapporteurs avant soutenance :

Mazen ALAMIR	Directeur de recherche CNRS, Université Grenoble Alpes (UGA), Gipsa-lab, UMR CNRS 5216
Hervé GUÉGUEN	Professeur des universités, CentraleSupelec, IETR, UMR CNRS 6164

Composition du Jury :

Président :	Eric BIDEAUX	Professeur des universités, INSA de Lyon, Ampère, UMR CNRS 5005
Examinateurs :	Mazen ALAMIR	Directeur de recherche CNRS, Université Grenoble Alpes (UGA), Gipsa-lab, UMR CNRS 5216
	Yacine GAOUA	Ingénieur-chercheur, CEA, CEA-Liten
	Hervé GUÉGUEN	Professeur des universités, CentraleSupelec, IETR, UMR CNRS
	Pauline KERGUS	Chargée de recherche, INP ENSEEIHT, Laplace, UMR CNRS 5213
Dir. de thèse :	Philippe CHEVREL	Professeur des universités, IMT Atlantique, LS2N, UMR CNRS 6004
Co-encadrant de thèse :	Fabien CLAVEAU	Maître de conférences, IMT Atlantique, LS2N, UMR CNRS 6004
	Pierrick HAURANT	Maître de conférences, IMT Atlantique, GEPEA, UMR CNRS 6144

Invité(s) :

Bruno LACARRIÈRE	Professeur des universités, IMT Atlantique, GEPEA, UMR CNRS 6144
Anthony MOURAUD	Innovation - R&D, Lhyfe
François NEPVEU	Ingénieur-chercheur, CEA, CEA Tech Pays de la Loire

Remerciements

Les travaux réalisés dans cette thèse de recherche ont été financés par la région Pays de la Loire avec une collaboration entre le CEA Tech Pays de la Loire, le laboratoire des Systèmes Energétiques pour les Territoires (LSET) du CEA-Liten, le laboratoire des Sciences du Numérique de Nantes (LS2N) et le laboratoire de Génie des Procédés–Environnement–Agro-alimentaire (GEPEA) de l'IMT Atlantique.

J'aimerai remercier le directeur de la thèse Philippe Chevrel pour les échanges, son implication et la rigueur scientifique pendant la réalisation de cette thèse de recherche. Je souhaite aussi remercier les personnes ayant participé à l'encadrement pour la bonne réalisation de ce travail : Fabien Claveau, Yacine Gaoua, Pierrick Haurant, Bruno Lacarrière, Anthony Mouraud et François Nepveu. Merci à Hervé Guéguen et Alexandre Goldsztejn pour la participation dans le Comité de Suivi Individuel (CSI) de la thèse de recherche.

Ensuite, je tiens à remercier l'ensemble du jury de thèse pour le temps passé à l'éxamination des travaux et de votre présence lors de la soutenance, Mazen Alamir et Hervé Guéguen rapporteurs des travaux, Pauline Kergus examinatrice et Eric Bideaux, Yacine Gaoua, Philippe Chevrel, Fabien Claveau, Pierrick Haurant examinateurs des travaux.

Ces années m'ont permis de rencontrer et de collaborer avec le personnel des laboratoires de recherche, je souhaite les remercier vivement.

Finalement, je souhaite remercier Andreea pour ses encouragements et son soutien pendant ces années.

SOMMAIRE

I Introduction générale 9

Introduction 11

1 Problématique de recherche et état de l'art 13

1.1 Problématique de recherche . . . 13
1.1.1 Energie . . . 13
1.1.2 Technologie de commande . . . 14
1.2 Etat de l'art . . . 15
1.2.1 Systèmes multi-énergies . . . 15
1.2.2 Pilotage de systèmes dynamiques . . . 18
1.2.3 Modélisation de systèmes dynamiques . . . 22
1.3 Positionnement du travail de recherche . . . 26
1.4 Conclusion . . . 28

II Élaboration d'une méthodologie 29

2 Méthodologie de commande 31

2.1 Synthèse . . . 32
2.2 Modélisation fine multiphysique . . . 34
2.3 Génération et récolte des données . . . 35
2.4 Paramétrisation du modèle boîte noire . . . 36
2.4.1 Prédiction des états du système . . . 36
2.4.2 Modèle boîte noire . . . 36
2.4.3 Entraînement et réglage des hyperparamètres . . . 44
2.4.4 Mesure de la fidélité . . . 47
2.4.5 Implémentation numérique . . . 47
2.5 Synthèse du régulateur . . . 48
2.5.1 Commande prédictive . . . 48
2.5.2 Implémentation numérique . . . 53

2.6 Evaluation du pilotage 54
2.7 Conclusion 54

3 Validation de la méthodologie 57
3.1 *Benchmarks* d'automatique 57
3.2 Application de la méthodologie 61
3.2.1 Modélisation fine multiphysique 61
3.2.2 Génération et récoltes des données 61
3.2.3 Paramétrisation du modèle boîte noire 67
3.2.4 Synthèse du régulateur 79
3.2.5 Evaluation du pilotage 82
3.3 Discussion 99
3.4 Conclusion 100

III Application de la méthodologie 101

4 Pilotage de la serre agricole 103
4.1 Production agricole sous serre 104
4.1.1 Serriculture 104
4.1.2 Serre agricole 106
4.2 État de l'art : pilotage de la serre 109
4.3 Présentation de la serre agricole de l'étude 112
4.4 Méthodologie 113
4.4.1 Modélisation de la serre agricole 113
4.4.2 Pilotage des variables climatiques 117
4.4.3 Pilotage classique des équipements supports 121
4.4.4 Pilotage avancé des équipements supports 124
4.4.5 Plan d'expérience 141
4.5 Résultats 147
4.5.1 Apprentissages des réseaux de neurones 147
4.5.2 Pilotage du système multi-énergie 155
4.6 Discussion 172
4.7 Conclusion 174

IV Conclusion et perspectives **175**

5 Conclusion **177**

6 Perspectives **179**
6.1 Perspectives technologiques . . . 179
6.2 Perspectives sur la commande . . . 180
6.3 Perspectives sur la prédiction du système dynamique . . . 181

V Annexes **183**

A Liste des figures **185**

B Liste des tableaux **193**

C Communications scientifiques **196**
C.1 Revues internationales . . . 196
C.2 Conférences internationales . . . 196
C.3 Présentations . . . 197
C.4 Rapports . . . 197
C.5 Enseignements . . . 197

D Mesure de la fidelité de prédiction pour la regression **198**
D.1 Erreur quadratique moyenne . . . 198
D.2 Racine de l'erreur quadratique moyenne . . . 198
D.3 Erreur moyenne absolue . . . 198
D.4 Erreur absolue moyenne en pourcentage . . . 199

E Réseaux de neurones non considérés dans cette étude **200**
E.1 Réseaux récurrents . . . 200
E.1.1 RNN . . . 200
E.1.2 Long short-term memory (LSTM) . . . 201
E.1.3 Gated recurrent unit (GRU) . . . 202
E.2 Réseaux dérivés de méthode de résolution numérique . . . 202
E.2.1 RKNN . . . 202
E.2.2 DenseNet . . . 203

E.2.3 FractalNet 203
E.2.4 LM-ResNet 205
E.2.5 OdeNet 206

F Modélisation détaillée de la serre agricole **207**
F.1 Bilan de puissance thermique 207
F.2 Bilan d'humidité 212
F.3 Bilan du dioxyde de carbone 213
F.4 Stockage thermique 214
F.5 Unité de production de chaleur 216

VI Références bibliographiques **219**

Bibliographie **221**

Première partie

Introduction générale

INTRODUCTION

Les systèmes multi-énergies consistent en l'agrégation de plusieurs vecteurs énergétiques comme l'électricité, la chaleur ou le gaz à travers des dispositifs de conversion d'énergie. Ils possèdent le potentiel pour l'intégration des énergies renouvelables aux réseaux de distribution afin d'améliorer leur efficacité (réduction des coûts d'opération et des émissions CO2). Les systèmes multi-énergies sont de plus en plus présents dans les installations industrielles telles que les usines, les centres commerciaux et les fermes agricoles. Pour exploiter ces systèmes au meilleur de leurs capacités, la loi de commande orchestrant les équipements relatifs aux différents vecteurs énergétiques est fondamentale. Son développement pose des défis fondamentaux portant sur les méthodes de modélisation et de synthèse. Le contexte général est le suivant :

- Systèmes multi-énergies : La mise en œuvre des systèmes multi-énergies est balbutiant dans le secteur agricole ;
- Algorithmes de commande : L'utilisation de commande avancée pour piloter les systèmes multi-énergies ont fait l'objet de peu d'études ;
- Modélisation : La modélisation des systèmes multi-énergies repose le plus souvent sur des équations algébriques simplifiées.

Ce mémoire s'intéresse en premier lieu au cas d'application constitué d'une serre agricole. Ce système contient les vecteurs électricité, chaleur et gaz avec un stockage de la chaleur et des consommations énergétiques associées. De plus, cette installation industrielle est fortement présente dans la région Pays de la Loire pour la production de légumes. Afin de répondre aux verrous technologiques identifiés, nous avons choisi de recourir à la méthode dite de commande prédictive. La commande prédictive nécessite un modèle dynamique représentatif du fonctionnement du système. Nous privilégierons une modélisation par prédiction de l'état, dont la fonction de prédiction est apprise par un réseau de neurones.

Ce mémoire comporte quatre chapitres qui présentent les différents points abordés précédemment :

- Chapitre 1 : il introduit la problématique de recherche et equisse un état de l'art ciblé des travaux sur les systèmes multi-énergies et le pilotage de systèmes dynamiques. Ensuite, le positionnement et la démarche scientifique sont exposés ;
- Chapitre 2 : il définit la méthodologie de commande envisagée, et son articulation avec

la modélisation multiphysique ;

- Chapitre 3 : ce chapitre illustre la pertinence de la méthodologie proposée sur un cas d'étude bien connu de la littérature ;
- Chapitre 4 : il décrit le cas d'application visé par cette thèse : la serre agricole. Son modèle mathématique est décrit, ainsi que le système multi-énergie support. Le système de pilotage hiérarchique conçu à partir de la méthodologie proposée est exprimé, avant d'être comparé avec le pilotage employé actuellement par les serristes ;
- Conclusion 5 La dernière partie du manuscrit présente un bilan des travaux exposés dans ce manuscrit, des contributions scientifiques et pour finir, des perspectives de recherche.

Ces travaux de thèse ont été financés par la région Pays de la Loire et encadrés d'une part par CEA Tech Pays de la Loire et le laboratoire des Systèmes Energétiques pour les Territoires (LSET) du CEA-Liten et d'autre part par le laboratoire des Sciences du Numérique de Nantes (LS2N) et le laboratoire de Génie des Procédés–Environnement–Agro-alimentaire (GEPEA) à l'IMT Atlantique.

Chapitre 1

Problématique de recherche et état de l'art

Résumé

Les systèmes multi-énergies se développent pour augmenter l'efficacité, la flexibilité et le déploiement optimal des sources. Pour piloter ces systèmes multi-énergies, des algorithmes de commande avancée sont nécessaires. Ce premier chapitre de la thèse présente, la problématique de recherche, ainsi que l'état de l'art pour le pilotage de ces systèmes multi-énergies. La problématique de recherche de la thèse est présentée dans la section 1.1, l'état de l'art est exposé à la section 1.2. Le positionnement du travail de recherche fait l'objet la section 1.3 tandis que la section 1.4 conclut le chapitre 1.

1.1 Problématique de recherche

La problématique de recherche est abordée sous deux aspects. Le premier à trait aux défis liés à la production énergétique et le second au pilotage des systèmes dynamiques.

1.1.1 Energie

Depuis la domestication industrielle de l'énergie par l'humain, la consommation d'énergie est en constante augmentation [1]. La production à grande échelle de l'énergie résulte de l'utilisation de pétrole, de charbon, de gaz, d'uranium, de chute d'eau, d'énergie solaire ou d'énergie cinétique du vent [2]. Par la même occasion, une large part de la production humaine d'énergie engendre des émissions de gaz à effet de serre et l'acroissement de la concentration de dioxyde de carbone dans l'atmosphère. Ceci est la cause de l'augmentation moyenne des températures de surface et de dérèglements climatiques [3].

Face à ces problématiques, la production et la distribution de l'énergie nécessitent de profonds changements [4]. En particulier, les ressources d'énergie décentralisée et renouvelable offrent la possibilité de réduire les dégâts causés à l'environnement [5]. Ces unités de production décarbonées sont : les systèmes solaires thermiques et photovoltaïques, les éoliennes, la microhydraulique au fil de l'eau. À cela s'ajoutent des systèmes de conversion énergétiques tels que les piles à combustible ou les pompes à chaleur. Ces nouveaux systèmes de production et de conversion engendrent des difficultés. Premièrement, ils dépendent de ressources naturelles intermittentes comme le soleil, le vent, la température extérieure ou le débit d'eau d'une rivière. Deuxièmement, ces systèmes sont de puissances réduites par rapport à une centrale conventionnelle. Troisièmement, les nouveaux systèmes de conversion réalisent le couplage entre plusieurs vecteurs énergétiques avec la chaleur, l'électricité ou l'hydrogène pour la pile à combustible.

Une nouvelle stratégie émerge avec les systèmes multi-énergies par la prise en compte des spécificités de ces productions énergétiques décarbonées. En particulier avec les différents vecteurs énergétiques disponibles et de l'intermittence des ressources. Les systèmes multi-énergies sont aussi parfois appelés polygénération. Ils combinent plus de deux systèmes de génération énergétiques et vecteurs énergétiques différents. L'objectif consiste alors à tirer le potentiel maximal de ces systèmes [5]. Les systèmes multi-énergies présentent des particularités qui les rendent intéressants [6] :

- Efficacité : L'énergie traditionnellement perdue devient utilisable et le facteur de charge des systèmes énergétiques est amélioré ; chaque ressource locale est optimisée ;
- Flexibilité : La flexibilité du système est perfectionnée ; les systèmes de stockage peuvent être développés et mieux utilisés ; la stabilité du réseau et l'intégration des sources de production d'énergies renouvelables sont améliorées ;
- Déploiement optimal : Le déploiement optimal des systèmes et des ressources énergétiques distribuées est facilité par les systèmes multi-énergies ;

Le potentiel des systèmes multi-énergies : efficacité, flexibilité et déploiement optimal, motivent notre recherche d'une solution de pilotage énergétique optimisée, rendue possible par l'utilisation de méthodes de commande élaborées.

1.1.2 Technologie de commande

Les systèmes modernes n'existeraient pas sans les technologies de commande, les avions, les fusées ou la robotique [7]. Le pilotage de ces systèmes amène des défis de recherche pour améliorer les performances, la sécurité, la fiabilité et réduire les coûts économiques. L'élabora-

tion d'une commande requiert quatre étapes. Tout d'abord, une analyse du système, puis sa modélisation sont réalisées. Ensuite, un algorithme de pilotage est conçu. Puis, la commande est le plus souvent validée dans un environnement de simulation, avant implémentation et tests expérimentaux [7], [8].

Des défis importants subsistent pour améliorer l'efficacité des principales ressources énergétiques renouvelables, des techniques de conversion, la fiabilité des systèmes et la prolongation de leur durée de vie [7]. De plus, les générateurs sont disséminés sur un territoire et les énergies primaires ne sont pas contrôlables (rayonnement, vents, rivières). L'intégration de ces énergies au niveau de la distribution énergétique nécessite des défis majeurs portant sur la commande notamment. Une large part de l'amélioration de l'efficacité provient d'arrêts d'équipements inutilisés ou la gestion des charges des processus industriels en accord avec la disponibilité des ressources énergétiques. La théorie de la commande est centrale pour l'optimisation du processus de production [7].

L'utilisation de technologie de commande pour piloter les systèmes multi-énergies est en mesure d'apporter des réponses aux problématiques soulevées et motive cette étude et ce travail de recherche.

1.2 Etat de l'art

Cette section présente un état ciblé articulant systèmes multi-énergies, pilotage et modélisation de systèmes dynamiques.

1.2.1 Systèmes multi-énergies

Il existe des systèmes qui réalisent le couplage entre vecteurs énergétiques. Ils effectuent une conversion d'un vecteur énergétique vers d'autres. Il existe aussi des systèmes dits de co-génération (chaleur et électricité). La pile à combustible réalise la conversion d'hydrogène pour produire de l'électricité et de la chaleur [9]. Cette approche peut être schématisée par la Fig.1.1.

Il existe aussi des systèmes de trigénération pour la production de froid, de chaleur et d'électricité. Dans ce cas, il est ajouté un second équipement (par exemple, une machine à absorption de production de froid) [6], Fig.1.2.

Le concept peut se généraliser en considérant un système avec plusieurs vecteurs énergétiques d'entrée et plusieurs en sorties. Dans ce cas, le système multi-énergie possède une

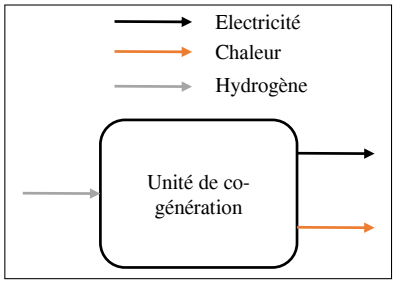

FIGURE 1.1 – Exemple d'une unité de co-génération.

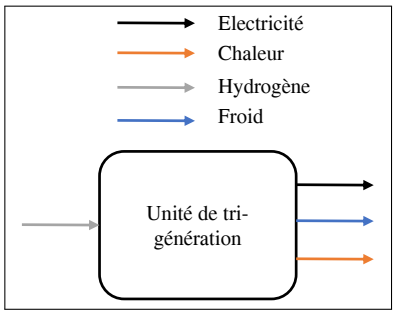

FIGURE 1.2 – Exemple d'une unité de tri-génération.

multitude de sous-systèmes réalisant le couplage entre deux ou trois vecteurs énergétiques [6], Fig.1.3.

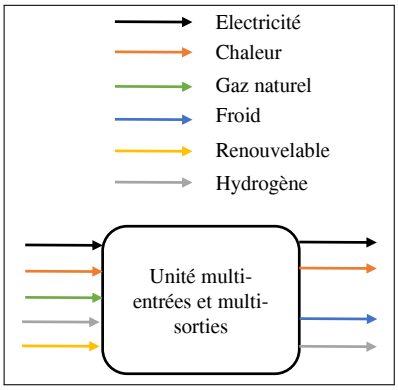

FIGURE 1.3 – Exemple d'une unité multi-entrées et multi-sorties.

Le couplage entre deux vecteurs énergétiques (chaleur et électricité) est étudié dans la littérature. À l'échelle locale dans l'étude [10] et à l'échelle d'un réseau dans [11]. Le document [10] présente les interactions de ces sous-systèmes énergétiques. Ces systèmes sont : un stockage électrique, la production électrique photovoltaïque, une pompe à chaleur, un stockage thermique. Les auteurs ont pris en compte touts les sous-systèmes pour obtenir des synergies

entre les vecteurs afin de réduire le coût économique global, et l'augmenter le confort de chauffage [10]. Par ailleurs, deux vecteurs énergétiques sont étudiés conjointement dans [11] pour dimensionner le plus efficacement possible les équipements d'un réseau de chaleur. Ces équipements sont : les pompes à chaleur et les chaudières, des panneaux photovoltaïques, des éoliennes et la connexion au réseau électrique. Les auteurs ont observé une réduction des coûts économiques par rapport aux dimensionnements des réseaux sans considérer les couplages des systèmes multi-énergies.

Le couplage entre trois vecteurs énergétiques est analysé dans [12]. Les vecteurs considérés sont : l'électricité, la chaleur et le gaz naturel. Les équipements de génération et de conversion sont : une unité de cogénération électricité et chaleur, un stockage thermique, des panneaux photovoltaïques, un stockage électrique. La demande énergétique est réalisée par des bâtiments résidentiels. L'étude est focalisée sur le pilotage du système. Les niveaux de pilotage sont séparés par les dynamiques temporelles entre dynamiques rapides et lentes. Les auteurs ont défini trois niveaux de pilotage, avec le premier niveau focalisé sur les dynamiques rapides (électricité), le second niveau sur les dynamiques moyennes (gaz naturel) et le dernier niveau pour les dynamiques lentes (chaleur).

Une étude concernant plus de trois vecteurs énergétiques est présentée dans [13]. Au-delà des vecteurs énergétiques, des produits agricoles (produits bruts et fertilisants) sont considérés. L'étude est focalisée sur la planification optimale des vecteurs et des équipements. Par ailleurs, l'usine agricole considérée a trait à la production d'huile d'olive [13]. Les vecteurs énergétiques étudiés sont : l'électricité, le gaz naturel, le bio gaz. Les produits agricoles pris en compte sont : les olives brutes, la production l'huile et les déchets valorisés comme fertilisant. Les équipements énergétiques utilisés pour l'étude sont : une unité de co-génération d'électricité et de chaleur, une chaudière, un transformateur électrique, des panneaux photovoltaïques, une connexion au réseau électrique et une pile à combustible. Les résultats de l'étude ont indiqué que la planification permet une réduction du coût économique par la prise en compte des variations tarifaires de l'énergie.

Dans cette section, nous avons passé en revue des systèmes multi-énergies avec plusieurs vecteurs énergétiques. Dans les sections suivantes, des méthodes de pilotage sont présentées pour piloter de tels systèmes.

1.2.2 Pilotage de systèmes dynamiques

Régulateurs classiques

Les régulateurs tirent leur intérêt d'un élément essentiel en automatique : la boucle fermée. Elle peut être présentée comme le retour d'information sur la commande par commentaires ou informations d'une action accomplie. Par exemple, le thermostat d'ambiance rétroagit sur la température, la chasse d'eau de toilette sur le niveau d'eau et l'homéostasie sur la température corporelle [14]. Il est usuel de schématiser la boucle fermée conformément à la Fig.1.4.

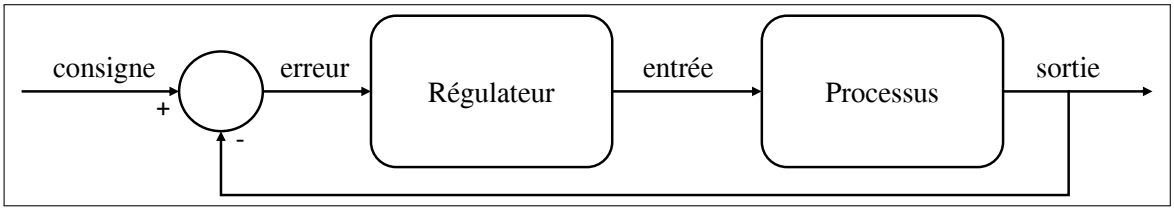

FIGURE 1.4 – Schéma d'une boucle fermée.

Le régulateur proportionnel, intégral, dérivé (PID) est le plus populaire des régulateurs pour piloter un système dynamique dans l'industrie. Il s'applique souvent sur la différence entre la consigne et la mesure de la variable à piloter, appelée l'erreur. La partie proportionnelle implique un simple gain, les parties intégrale et dérivée des gains sur l'intégrale et la dérivée d'erreur [15]. Une approche fréquentielle est parfois mise à profit pour obtenir une stabilisation et une certaine robustesse [16]. Un second régulateur encore plus simple est le régulateur Bang-Bang. Il possède deux positions : fermé ou ouvert. Les principaux avantages sont une simplicité de mise en oeuvre et un coût réduit. Cependant, les possibilités de stabilisation ou de robustesse du système sont réduites. Un régulateur dit à base de règle ou *rule-based* procède à base de test conditionnels pour agir sur les actionneurs à partir des sorties du système à piloter.

Ces régulateurs classiques sont simples de mise en oeuvre et nécessitent peu de connaissances. Néanmoins, ils sont insuffisants pour piloter de manière optimisée des systèmes multivariables aux dynamiques couplées. Nous considérons donc, dans ce qui suit, des commandes parfois qualifiées d'avancées.

Commande avancée

Afin de faciliter la conception d'une loi commande, une voie consiste à la décomposer en plusieurs niveaux hiérarchiques. Chaque niveau assure une fonction particulière, comme calculer une trajectoire de référence pour le niveau hiérarchique inférieure. Le principe de

la commande hiérarchique est ancien. En 1970, un article proposait un état de l'art sur ce sujet [17]. Il y est noté que l'optimisation d'un système est difficile et qu'il est préférable de séparer conceptuellement l'optimisation/la planification de la régulation. Le développement d'une commande hiérarchique scindant la planification optimisée et la régulation permet de résoudre cette difficulté [1]. La commande hiérarchique peut possèder trois niveaux de pilotage. Le premier niveau est alors responsable de la régulation, le second niveau de la planification et le troisième niveau de la coordination. Un schéma général de la commande hiérarchique est représenté à la Fig.1.5. D'autres bénéfices peuvent être attendus de la commande hiérarchique. Par exemple, la réduction de la complexité mathématique et numérique de la commande [20]. Dans [21], la commande hiérarchique est dédiée pour deux scénarii : la coordination de système et le pilotage avec plusieurs niveaux. La commande hiérarchique est appropriée pour le pilotage multiniveaux quand plusieurs régulateurs travaillent de concert avec des échelles de temps séparées. La commande hiérarchique permet de prendre en compte des échelles de temps de la seconde à plusieurs années, ainsi chaque niveau est dédié à une échelle de temps spécifique [22]. De surcroît, la commande hiérarchique autorise l'intégration du pilotage avec branchement et utilisation rapide (*plug-and-play*) [23]. Elle permet aussi l'amélioration de la robustesse du pilotage par une plus grande tolérance aux défauts [24].

Les applications de la commande hiérarchique sont nombreuses. La nature l'utilise par exemple, pour la réalisation du chant d'un oiseau [25], de la température de système biologique [26] ou de l'organisation humaine [27]. Néanmoins, les systèmes qui retiennent notre attention dans ce travail de recherche concernent les systèmes industriels construits par les humains et plus encore les systèmes énergétiques et multi-énergies. Par exemple, une commande hiérarchique est utilisée pour piloter une succession de centrales hydroélectriques selon des bases tarifaires avec un niveau de coordination et des niveaux de régulation [28], [29]. La commande hiérarchique est aussi utilisée pour piloter des éoliennes et des stockages par volant d'inertie [30], une centrale solaire [31], des centrales conventionnelles [32] ou un microréseau électrique [33].

Le second niveau de pilotage de la commande hiérarchique retient notre attention pour le travail de cette thèse de recherche. Ce niveau s'appuie sur le principe de la commande optimale, dont la consigne est transmise par le niveau supérieur.

La commande optimale émergea dans les années 1960, dans le contexte de la représentation d'état des systèmes dynamiques. Cette période est considérée comme l'âge d'or de l'automatique

1. Il est souhaitable d'indiquer que les capacités informatiques de calculs ont progressé depuis 1979. Début de la décennie 1970 Intel sortait le premier microprocesseur commercial le 4004 avec 2 300 transistors [18], actuellement les processeurs modernes dépassent 10^{10} transistors [19].

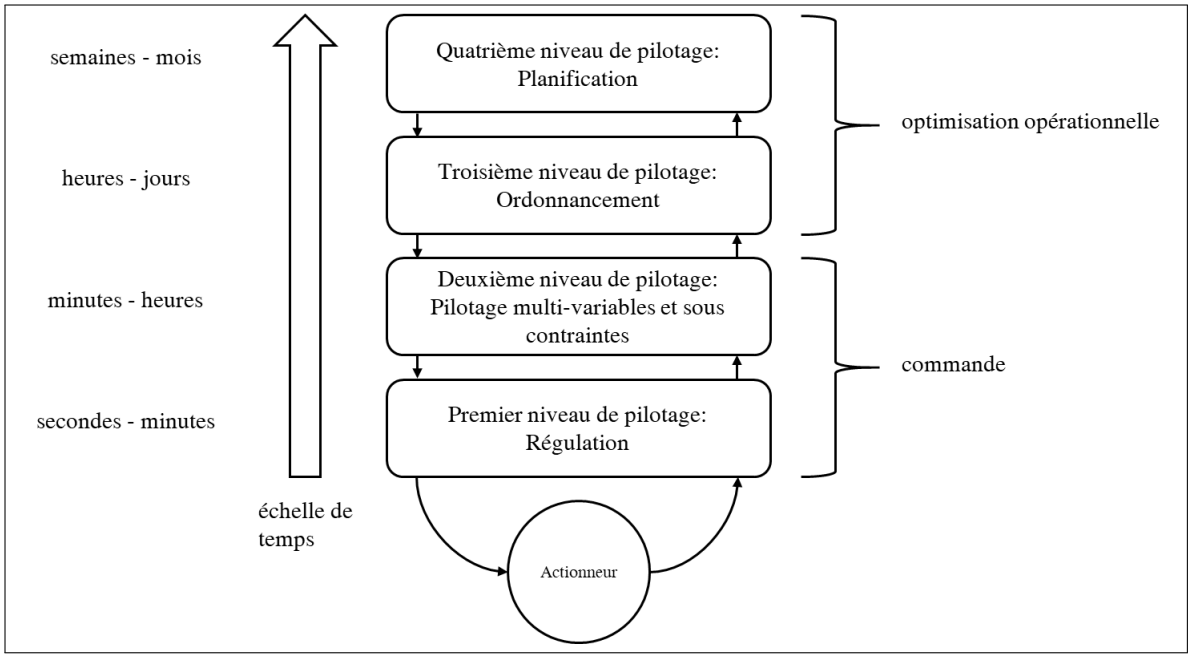

FIGURE 1.5 – Schématisation d'une commande hiérarchique, adapté de [34].

avec les travaux de Bellman, Kalman, Pontryagin et autre [35]. Les applications concernaient les industries spatiales et les missiles balistiques [35]. L'idée centrale de la commande optimale réside dans la minimisation d'une fonction coût avec ou sans contraintes. La commande à critère quadratique se distingue, par l'utilisation d'une fonction coût faisant intervenir l'erreur de consigne et l'énergie de commande requise. Elle possède la capacité à optimiser le comportement dynamique d'un système multivariable et le pilotage de système avec plusieurs entrées et plusieurs sorties [36]. Lorsque ce type de commande optimale ne possède pas de contraintes, le régulateur est appelé *Linear Quadratic Regulator* (LQR) et une équation algébrique de Riccati est résolue pour obtenir les paramètres du régulateur [36]. Un exemple d'application de la commande optimale est visible dans le document [37], avec le pilotage de la direction latérale d'un véhicule. La commande prédictive reprend le concept de commande optimale et utilise pour cela classiquement un critère quadratique à horizon fini auxquelles peuvent être ajoutées des contraintes sur les états et les commandes. La commande optimale est alors recalculée à chaque changement d'état du processus considéré, afin d'actualiser la commande [38].

La première implémentation de la commande prédictive (abréviation anglaise MPC pour *Model Predictive Control*) a été réalisée pour piloter des systèmes de l'industrie pétrolière par Richalet *et al.* en 1978 [39]. Les éléments caractéristiques de la MPC sont les suivants : i) utilisation d'un modèle représentatif du comportement du processus, ii) fonction quadratique

sur les états et les entrées, iii) prise en compte des contraintes sur la trajectoire des états et entrées, iV) fonction de coût et/ou contraintes terminales sont parfois utilisées afin d'améliorer les conditions de stabilité et de faisabilité.

L'utilisation de la commande MPC pour piloter un système dynamique est répandue et il est impossible de réaliser un inventaire exhaustif des applications. Néanmoins, nous avons retenu quelques applications. Nous avons retenu qu'elle est utilisée pour le guidage de véhicule spatial [40], pour le pilotage de feux tricolores [41], ou bien pour le pilotage de voitures de course autonomes [42]. En outre, la commande MPC intervient parfois dans le cadre d'une une commande hiérarchique [43]. Dans [43], une commande à deux niveaux est utilisée, dont le premier niveau est constitué d'une commande prédictive qui reçoit les consignes d'un second niveau constitué d'une commande prédictive économique. Dans le cadre des systèmes énergétiques, la MPC est utilisée pour piloter un système contenant une pile à combustible, une batterie et des supercondensateurs [44]. En outre, il est proposé dans [45] l'utilisation d'une commande hiérarchique contenant deux MPC. Une première pour la planification et une seconde pour le pilotage du système énergétique d'une maison résidentielle. Dans le domaine résidentiel, nous pouvons citer [46] avec une MPC contenant une fonction coût économique pour piloter la ventilation de plusieurs bâtiments d'un campus universitaire. Par ailleurs, la MPC avec une fonction de coût économique a aussi été appliquée [47] pour piloter un système électrique avec de la production électrique, du stockage et des charges programmables. Le pilotage de la demande énergétique est aussi considérée dans le document [48]. L'objectif visé est la réduction des surcharges électriques. Pour cela, les auteurs ont modélisé une habitation avec un modèle résistance-capacité. Ensuite, le pilotage est assuré par une commande MPC multiobjectif visant la minimisation de l'inconfort de la température intérieure et de la puissance consommée par le système.

Commandes adoptées pour piloter les systèmes multi-énergies

Dans cette section des exemples de pilotage de système multi-énergies sont tirés de la littérature. Tout d'abord l'utilisation de l'organisation hiérarchisée de la commande a été employée pour améliorer la coordination entre différents vecteurs énergétiques. Une meilleure intégration technique, économique et environnementale est obtenue par rapport au pilotage des vecteurs séparés [6]. Ensuite, un pilotage d'un système multi-énergies avec plusieurs niveaux de commande incluant un niveau d'optimisation et un niveau de régulation est présenté dans la littérature [49]-[51]. Dans [49], le pilotage d'un micro-réseau est réalisé par une commande hiérarchique contenant une commande prédictive (MPC) adaptée à différentes échelles de

temps. La commande possède deux échelles de temps. Le premier niveau s'occupe de la question de la stabilisation et le second de la gestion énergétique. Le second article [50] traite d'une commande hiérarchique avec deux niveaux, respectivement dénommés commande unitaire et commande principale. La commande unitaire est composée de contrôleurs locaux qui reçoivent les consignes de la commande principale. Finalement dans [51] les auteurs introduisent trois niveaux de pilotage. Le premier niveau réalise la coordination entre les vecteurs énergétiques, le second s'occupe d'optimisation et le dernier réalise la régulation en boucle fermée.

La commande MPC est parfois utilisée pour piloter un système multi-énergie contenant plusieurs vecteurs énergétiques. Dans le cadre d'une maison résidentielle, une commande MPC concerne la gestion de deux vecteurs énergétiques (électricité et chaleur) [52]. L'utilisation d'une commande MPC avec une fonction coût économique pour le pilotage combiné de système énergétique thermique et électrique est présentée dans [53]. Le système énergétique est constitué de stockage thermique et électrique, de production électrique par panneaux photovoltaïques et de deux convertisseurs du vecteur électrique vers la chaleur. Le modèle de la MPC est linéarisé à partir d'un modèle non linéaire, les auteurs ont obtenu une réduction du coût économique par rapport à un pilotage réalisé par régulateur PID. Un micro-réseau énergétique avec une intégration de système thermique et électrique est considéré dans [54], les auteurs ont utilisé la commande MPC pour piloter le système en regard de tarifs électriques. Un système multi-énergies constitué de vecteurs gaz naturel, électricité et chaleur est piloté avec la commande MPC dans [55]. La modélisation du système est enfin réalisée par la formulation d'un bilan de puissance et la fonction coût prend en compte les coûts économiques de l'électricité et du gaz naturel.

1.2.3 Modélisation de systèmes dynamiques

Modèles ?

Les régulateurs considérés dans ce travail de thèse s'appuient sur un modèle dynamique, noté g, du système à contrôler. Définir un modèle dynamique du système à piloter est une tâche délicate. Henri Poincaré indiquait que la vraie nature des choses ne peut-être décrite par la science et que la science est une classification qui est une description humaine commode, mais n'indique pas la vraie nature des relations [56]. Dans ce qui suit, la "vraie nature" du système sera notée par f [2]. Une définition de ce qu'est un modèle est indiquée dans le livre de Peter Fritzson [57] : "Un modèle d'un système est une expérience qui peut être appliquée dans le but

2. f est le système réel et g est le modèle

de répondre aux questions sur le système". Le mot expérience a besoin d'être spécifié dans ce cas. L'auteur indique : "Une expérience est un processus d'extraction de l'information sur le système en excitant ses entrées". Par ailleurs, l'auteur classe plusieurs possibilités de modèle :

- Modèle mental : il est une affirmation qui permet de répondre à une question à propos d'un comportement. Par exemple, l'auteur utilise la phrase : "Une personne est fiable".
- Modèle verbal : ils sont décrits par des phrases et des mots. L'auteur donne l'exemple suivant : "Plus d'accidents surviennent si la vitesse limite est augmentée".
- Modèle physique : ils sont des répliques de système réel permettant la mise en évidence de caractéristiques particulières du système réel. Nous pouvons citer l'exemple du modèle d'avion de taille réduite testé dans une soufflerie.
- Modèle mathématique : ils expriment des relations entre variables. Des exemples de modèles mathématiques sont les équations de Maxwell [58] ou les lois du mouvement de Newton [59].

Les modèles qui retiennent l'attention ici sont les modèles mathématiques formulés avec des équations mathématiques, des fonctions mathématiques et des programmes informatiques [57]. Les modèles mathématiques peuvent être catégorisés entre trois groupes selon le niveau de détail des équations physiques ou de la méthode d'obtention : modèle boîte blanche, modèle boîte grise, modèle boîte noire [60] :

- Modèle boîte blanche : le modèle décrit en détail le système dynamique et toutes les équations et les paramètres physiques sont connus.
- Modèle boîte grise : le modèle a une connaissance incomplete des équations physiques ou des paramètres.
- Modèle boîte noire : Les équations physiques ne sont pas connues et elles ne sont pas utilisées. La dynamique du système est obtenue par les mesures d'entrées et sorties, puis un traitement est réalisé sur le modèle pour le paramétrer.

Dans ce travail, nous avons priviliégié en partie les modèles boîtes noires. Ces modèles permettent d'utiliser des données de mesure du système et de réduire le temps d'ingénierie pour l'obtention d'un modèle [61].

Modèles boîtes noires

Les données d'entrée et de sortie du système sont utilisées pour paramétrer un modèle mathématique sans lien avec des équations physiques. Puis, les paramètres du modèle sont accordés pour obtenir des signaux de sortie les plus fidèles possible par rapport aux données de sortie. Une grande variété de modèles est envisageable. Les modèles paramétriques du

type *AutoRegressive Exogenous (ARX)* est par exemple utilisé comme support pour la MPC dans [62]. Les modèles à base de neurones artificiels également [63] qui sont utilisés pour la commande MPC dans [64] ou la commande EMPC dans [65]. Nous avons dans ce travail retenu les modèles à base de neurones artificiels pour leur potentiel à améliorer la prédiction des systèmes dynamiques par rapport à des modèles paramétriques [66]. Les réseaux de neurones sont utilisés pour des cadres variés. A tire d'exemple, ils sont utilisés pour réaliser la prédiction de consommation énergétique d'un bâtiment [67], le traitement automatique de la parole [68], la reconnaissance de forme [69] ou bien l'approximation de fonctions [70]. Dans notre recherche, les neurones artificiels sont utilisés pour réaliser la prédiction du comportement du système dynamique visé, Fig.1.6.

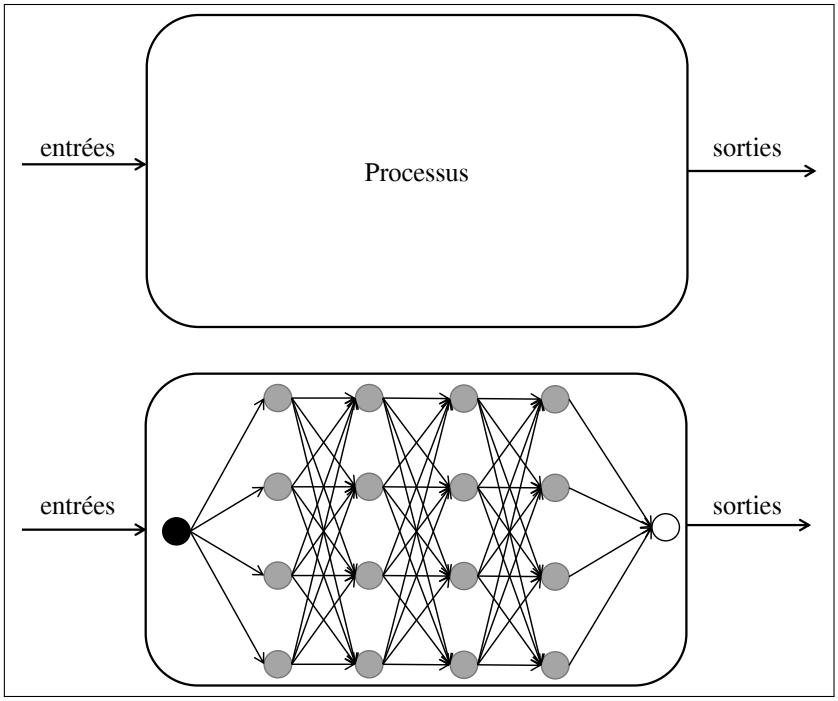

FIGURE 1.6 – Modèle boîte noire d'un système dynamique.

Différentes architectures neuronales sont utilisées pour la prédiction d'un système dynamique : le perceptron multicouche dans [71] et aussi le perceptron multicouche avec entrées retardées : *Time delay neural network* dans [72] pour la prédiction d'un modèle dynamique d'un motard. Des neurones récurrents sont aussi utilisés pour prédire le système dynamique et utilisé avec une commande MPC pour piloter un processus chimique non linéaire [64]. D'autres neurones récurrents tel que *Long short-term memory* (LSTM) et *Gated recurrent unit* ont été utilisés à des fins de prédiction de systèmes dynamiques [73], [74]. Ils ont été utilisés avec la

commande MPC pour piloter un système non linéaire [66], [75].

Modèlisation de système multi-énergies

Un nombre conséquent de travaux de recherche utilise le concept de hub énergétique pour réaliser la modélisation de systèmes multi-énergies. La modélisation du système multi-énergie par le hub énergétique s'est révélée appropriée pour de nombreuses installations industrielles. Citons le cas des usines à papiers et les aciéries, de grandes installations (aéroports, hôpitaux, grandes surfaces marchandes), des systèmes urbains et ruraux à plus petite échelle (trains, bateaux, avions) [76]. Le hub énergétique a été proposé pour la première fois en 2005 par un groupe de personnes de *Power Systems and High Voltage Laboratory* de l'ETH de Zurich [77]. Le hub énergétique a la capacité de connecter les productions d'énergie, les demandes, les éléments de stockage, les systèmes de transmission et de conversion [78]. Une illustration d'un schéma d'un hub énergétique est visible à la Fig.1.7. Le hub énergétique présenté contient un transformateur électrique, un stockage électrique, une unité de co-génération, une chaudière au bois, un stockage de la chaleur et une machine à absorption. Par ailleurs, les entrées énergétiques concernent l'électricité, le gaz naturel, la chaleur et les bûches de bois. En outre, les demandes énergétiques sont effectives sur l'électricité, la chaleur et le froid.

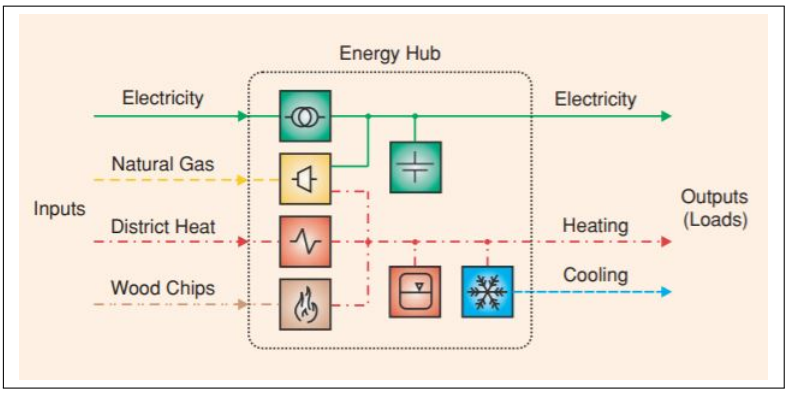

FIGURE 1.7 – Exemple d'un hub énergétique, tiré de [79] ©[2011] IEEE.

La modélisation d'un système multi-énergies avec la formulation par le hub énergétique utilise des équations mathématiques, il en résulte un modèle boîte grise. Il repose sur le principe de flux énergétique. Les flux d'entrée qui représentent les sources (gauche sur la Fig.1.7) et les flux de sortie avec les demandes énergétiques (droite sur la Fig.1.7). Entre les deux, sont renseignés les équipements réalisant les conversions énergétiques et le stockage. Ainsi, l'élément essentiel est une matrice de couplage entre les différents vecteurs pour obtenir un modèle

mathématique tout en prenant en compte les stockages énergétiques. En outre, cette méthode de modélisation est bien appropriée pour obtenir les équations algébriques du système.

Les études se basant sur la modélisation du système multi-énergies par le hub énergétique sont nombreuses [80]. Par exemple, il est utilisé pour l'analyse des flux de puissance d'un système couplé entre un réseau électricité et un réseau de chaleur [81]. La modélisation par le hub énergétique est aussi utilisée pour réaliser une optimisation et la planification de système multi-énergies [82].

La modélisation d'un système multi-énergie a aussi été abordée dans des travaux antérieurs à cette thèse de recherche lors du stage de master [83]. Le travail avait pour but d'améliorer la modélisation d'un système multi-énergie par rapport à la méthode du hub énergétique :

- Le hub énergétique permet la modélisation par des équations algébriques, le travail a consisté à prendre en compte des équations récurrentes du système multi-énergie pour observer les dynamiques temporelles du système multi-énergie.
- Le hub énergétique réalise une distinction forte entre les demandes et les sources, Fig.1.7. L'amélioration a consisté à prendre en compte des équipements énergétiques avec la faculté de consommation ou de restitution énergétique aux réseaux.

Néanmoins, ce travail présente toujours quelques limitations, en particulier les équations récurrentes sont présentées à partir d'équations linéaires du premier ordre.

1.3 Positionnement du travail de recherche

Le travail de la thèse s'articule autour de trois problématiques de recherche pour proposer des réponses aux défis présentés dans la section 1.1. La considértion des trois simultanément est un élément de différenciation fort par rapport aux travaux antérieurs. En outre, l'état de l'art présenté en section 1.2 a mis en lumière des verrous technologiques :

- multi-énergies : La prise en compte des différents vecteurs énergétiques apporte un potentiel d'amélioration de l'efficacité de conversion énergétique, de la flexibilité au système énergétique et un déploiement optimal des systèmes. Néanmoins, beaucoup reste à faire pour observer ces notions sur des systèmes énergétiques un peu atytiques possédant plusieurs vecteurs énergétiques. De nombreux systèmes multi-énergies ont été étudiés dans la littérature pour les bâtiments résidentiels ou les micro-réseaux. Dans le même temps, les études relatives à d'autres cadres sont encore balbutiantes. Compléter l'existant en mettant en évidence les particularités et les avantages des systèmes multi-énergies sur d'autres systèmes reste un enjeu.

- Algorithmes de commande : Un certain nombre d'études qui traitent de la planification d'un système multi-énergies. Ces planifications sont basées avec des équations algébriques et la minimisation d'une fonction de coût économique. L'utilisation d'une commande avancée a été peu considérée dans la recherche sur les systèmes multi-énergies, s'appuyant sur des équations récurrentes *ad hoc*. Le développement de méthodes de commande avancées et génériques constitue un enjeu fort pour l'avenir.
- Modélisation : Les principales modélisations des systèmes multi-énergies rencontrées dans la littérature s'appuient sur des modélisations linéaires et invariantes dans le temps extrait des équations mathématiques. Cette méthode engendre deux difficultés. Tout d'abord, il est nécessaire de posséder une certaine connaissance à priori du système. Ensuite, la modélisation est simplifiée et ne prend pas en compte certains éléments de réalité qui peuvent s'avérer importants. L'obtention du modèle un tant soit peu précis d'un système multi-énergie, sans en avoir une connaissance préalable approfondie, constitue un enjeu important au regard de l'objectif de cette thèse.

En relation avec ces différents enjeux et défis, cette thèse effectue des choix méthodologiques relatifs à la modélisation et la commande, de manière à répondre à la problématique complexe de la gestion multi-énergie d'une serre agricole :

- Serre agricole : La serre agricole avec les équipements énergétiques supports est un système multi-énergies avec les vecteurs électrique, chaleur, gaz naturel et dioxyde de carbone comme fertilisant de la plante.
- Commande avancée : Pour piloter les systèmes multi-énergies, une commande hiérarchique à deux niveaux est privilégiée. Le premier réalise la planification et le second élaborre la commande respectant les contraintes du système. Le premier niveau réalise une optimisation des coûts économiques et le second étage est basé sur la commande MPC. Cette commande considère plusieurs entrées et sorties, les contraintes sur les actionneurs et les états du système.
- Modèle boîte noire : Nous avons retenu pour la modélisation de systèmes multi-énergies, une modélisation détaillée sans connaissance à priori des équations physiques. Cette capacité des modèles boîtes noires permet de réduire le temps d'ingénierie pour l'obtention d'un modèle. En outre, les modèles boîtes noires choisies sont basés sur des neurones artificiels, le modèle neuronal retenu permet de prédire les sorties de système dynamique complexes.

Le travail de la thèse est structuré en deux parties et trois chapitres (cf. Fig.1.8) :

- Élaboration d'une méthodologie : Élaboration de la synthèse d'un modèle de prédiction

efficient (compromis qualité de prédiction et coût de calcul), à celle d'une commande MPC embarquant ce modèle. Cette partie comporte deux chapitres. Le chapitre 2 présente la méthodologie de synthèse de la commande MPC à base de process physique voire modèle multi-physiques approché par un réseau de neurones. Le chapitre 3 apporte une validation de principe de cette méthodologie sur la base d'un *benchmark* multivariable très utilisé en automatique.

— Application de la méthodologie : Fort de la validation du principe méthodologique au chapitre 3, le chapitre 4 procède à sa mise en application sur le système multi-énergie d'une serre agricole dont les caractéristiques sont spécifiées par CEA Tech.

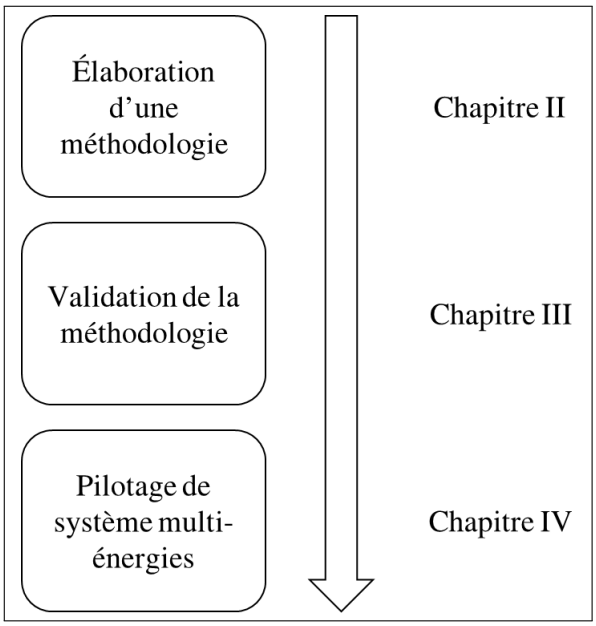

FIGURE 1.8 – Plan de la thèse de recherche.

1.4 Conclusion

Ce chapitre a introduit les problématiques de recherche de la thèse, questionnant la modélisation et le pilotage de systèmes multi-énergies complexes. Pour bien l'appréhender, un état de l'art lui a été associé, qui traite de manière ciblée des systèmes multi-énergies, du pilotage de systèmes dynamiques et modélisation neuronale de systèmes dynamiques. Le plan de la thèse clôt ce chapitre introductif.

Deuxième partie

Élaboration d'une méthodologie

Chapitre 2

Méthodologie de commande

Résumé

Le développement d'une méthode de pilotage à base de commande prédictive nécessite de disposer d'un modèle de représentation du processus capable de prédiction. Un tel modèle mathématique servira alors de support à la personne en charge de la conception de l'algorithme de pilotage. Néanmoins, trouver ces équations pour des systèmes d'un certain niveau de complexité nécessite un temps d'ingénierie important de modélisation qui peut être incompatible avec les attentes industrielles [61]. Une seconde méthode est de paramétrer un modèle boîte noire en utilisant les données du système physique, appréhendé comme un système dynamique. Toutefois, dans le processus de management d'ingénierie il est intéressant de concevoir la commande à mettre en place sur un système avant sa réalisation physique [84].

Ce chapitre vise à définir une démarche méthodologique suffisamment générale conduisant au pilotage optimisé d'un système dynamique donné. Cette méthodologie procède à partir de différentes briques technologiques. Elle s'appuie, pour commencer, sur la modélisation fine de systèmes dynamiques. Cette modélisation est considérée comme une réplique numérique d'un processus, réalisée avec des outils spécifiques. Elle recourt ensuite aux méthodes d'apprentissages profonds à partir de données pour paramétrer un modèle de type boîte noire. Elle fait appel enfin à la commande prédictive dont la synthèse est réalisée sur la base du modèle boîte noire. Ce chapitre présente chacun des aspects.

Une partie des travaux de ce chapitre ont fait l'objet de la rédation de deux articles. Le premier dans une revue [85] et le second article est en cours de modification après une première relecture par les pairs [86].

La section 2.1 présente une vision macroscopique de la méthodologie mise en place dans ce travail de recherche. Ensuite, la modélisation fine est exposée en section 2.2, la génération des données est discutée en section 2.3. La section 2.4 expose la paramétrisation du modèle boîte noire. La synthèse de la commande prédictive est décrite en section 2.5, tandis que son évaluation est présentée en section 2.6. La section 2.7 conclut ce travail méthodologique.

2.1 Synthèse

Cette synthèse donne une vue macroscopique de la méthodologie proposée afin d'en simplifier la lecture. La figure 2.1 notamment décrit les différentes étapes détaillées ci-après. Le point de départ est bien sûr de considérer le pilotage d'un système dynamique par une commande avancée optimale. Les différentes étapes présentées par la Fig.2.1 sont :

- Modélisation fine multiphysique : Une modélisation fine du système dynamique à piloter est effectuée. En premier lieu, cette modélisation peut être effectuée à l'aide du langage de modélisation multiphysique tel que Modelica. Il est intéressant de s'appuyer sur le travail de spécialiste en modélisation et des bibliothèques de composants disponibles.
- Génération et récolte des données : Il s'agit lors de cette deuxième étape de réaliser des simulations numériques, afin de générer et récolter les données qui permettront la calibration du modèle boîte noire. Durant les simulations, des signaux sont appliqués aux actionneurs pour révéler les particularités fréquentielles et dynamiques du système numérique. De plus, un prétraitement et mise en forme des données peuvent être pratiqués. Cette seconde étape motive l'utilisation d'un modèle multiphysique fin ; en effet, même si le système à piloter existe déjà physiquement, il est parfois difficile de générer toutes les données nécessaires, pour des raisons de coûts, de temps, de disponibilité, de sécurité....
- Paramétrisation du modèle boîte noire : Le modèle boîte noire est paramétré et ces paramètres constituent les variables de décision du problème d'optimisation guidant l'apprentissage. La fonction objectif rend compte du degré de similarité entre les données et les signaux prédits. Cette mesure permet d'afficher la qualité de reproduction des signaux.
- Synthèse du régulateur : La synthèse du régulateur optimal est réalisée. La synthèse du régulateur utilise le modèle boîte noire, tout en prenant en compte les capacités de la commande MPC : contraintes, prédictions, stabilisation.
- Évaluation du pilotage : La commande MPC est évaluée numériquement pour conclure sur les performances atteignables. Les indices de performance rendent compte du respect des contraintes, des temps de calcul entre chaque itération. En outre, une comparaison avec des régulateurs simples tels que *rule-based* ou PID peut compléter l'analyse si ceux-ci pré-existent. Si les indices de performance considérés sont correctement respectés, l'application du régulateur en vue du pilotage du système réel est envisagée [1].

La méthodologie présentée possède une certaine généricité qui permet l'application sur

1. Cette application sur système réel sort du travail de la thèse de la recherche. L'application du régulateur finale et les étapes de préparation associées sortent du travail de cette thèse.

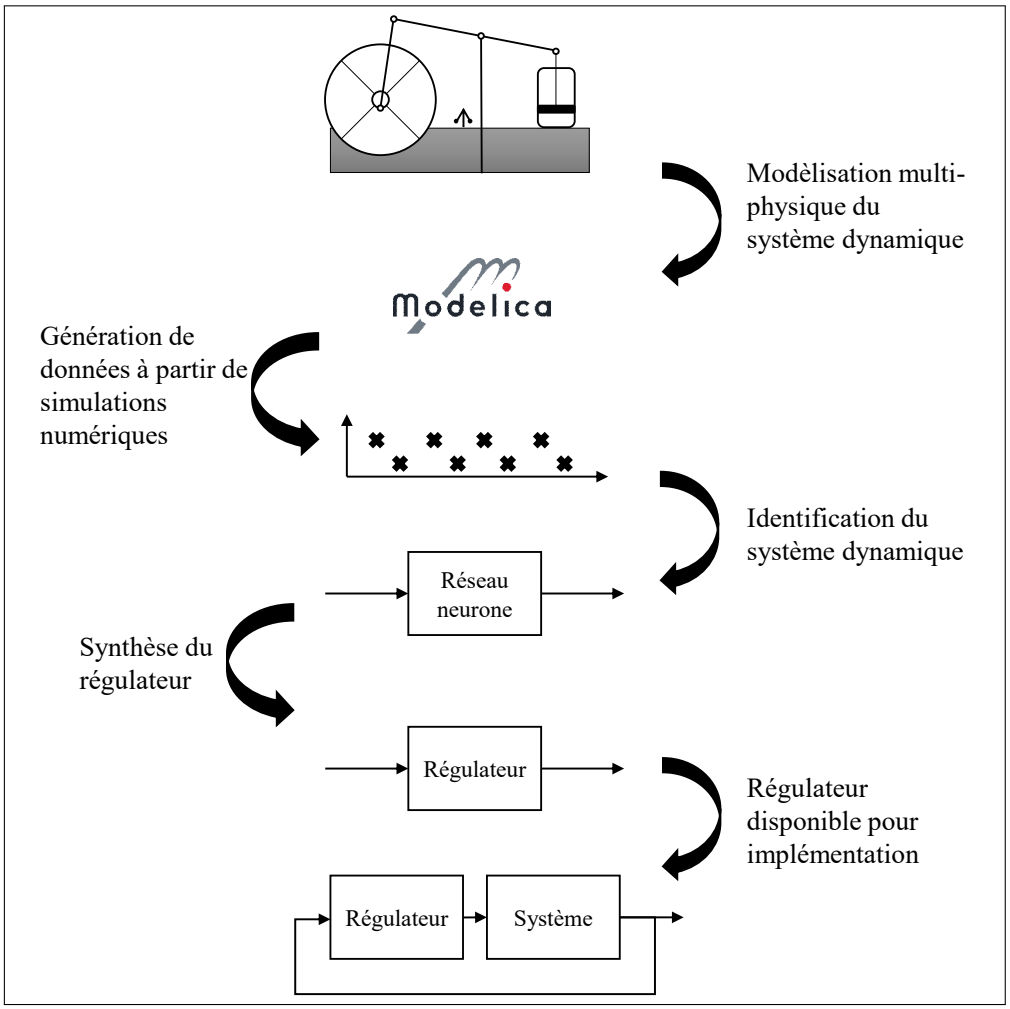

FIGURE 2.1 – Schématisation de la méthodologie proposée.

plusieurs systèmes dynamiques. Néanmoins, des contraintes doivent être respectées, en particulier :

- Le système dynamique doit être stabilisable.
- Touts les états sont disponibles pour la commande optimale.
- Le système dynamique doit être stable avant l'application de la méthodologie : stable par nature ou stabilisé par un régulateur bas-niveau. La mise en place de ce régulateur n'est pas présentée dans ce document.

Les différentes étapes décrites dans cette synthèse sont présentées en détail dans les sections suivantes.

2.2 Modélisation fine multiphysique

Dans cette section, nous abordons la modélisation fine multiphysique. Différentes approches existent, mais nous nous contenterons de l'approche en 0-D. La modélisation 0-D est réalisée avec des outils informatiques, par exemple Simscape de Mathworks [87], Modelica [88] ou Modia [89]. Sous l'hypothèse que tout système physique peut se décomposer en chaînes d'énergie et chaîne d'information, la modélisation multiphysique apporte des outils et des méthodes permettant de modéliser les flux d'énergie. Pour cela, les logiciels multiphysique permettent ; 1/ de décomposer un système complexe ou sous-systèmes élémentaires bien identifiés, 2/ de modéliser de manière à normaliser leur interaction via les variables énergétiques, typiquement le couple (effort, flux).

De plus, la grande force de ces logiciels est de mettre à disposition des librairies réalisées par des experts d'un domaine (thermique, électrique, mécanique...) proposant un ou plusieurs fins d'un organe donné (e.g. batterie électrique, unité de co-génération, chaudière, etc...). L'ingénieur peut alors utiliser ces modèles d'organe élémentaires sans avoir à maîtriser parfaitement les équations qui les régissent, lui permettant de se focaliser sur l'architecture globale du système complet à modéliser. Ces librairies (commerciales ou open sources) sont alimentés par une communauté de plus en plus importante, issus du monde académique ou industriel. Par exemple, dans le domaine des systèmes purement énergétiques qui nous intéresse dans cette thèse, la librairie Buildings [90] ou BuildSysPro [91] proposent des modèles et sous-modèles pour la modélisation thermique du bâtiment. La librairie Greenhouses [92] autorise la modélisation de serre agricole, la librairie DistrictHeating [93] pour les réseaux de chaleur, ou bien la librairie TransiEnt [94] permet la modélisation de systèmes énergétiques. Dans ce travail de recherche, nous avons sélectionné le langage Modelica [88], car un nombre important de librairies sont

disponibles et le CEA développe la librairie DistrictHeating [93].

2.3 Génération et récolte des données

Modelica [88] est un langage de programmation à part entière. Afin de créer et exécuter le programme Modelica, un environnement de simulation est requis. Il existe plusieurs logiciels pour réaliser la modélisation des processus avec ce langage, par exemple Dymola de Dassault Systems [95] ou bien OpenModelica [96]. Dans ce travail nous avons sélectionné Dymola. La modélisation multiphysique d'un processus engendre des équations différentielles et algébriques. Dans le but d'intégrer et résoudre ces équations, des solveurs appropriés sont nécessaires, citons pour exemple, le solveur Differential/Algebraic System Solver (DASSL) [97], [98] [2] ; ce solveur est multi-pas, il est intégré dans le logiciel Dymola et OpenModelica et il sera utilisé dans le cadre de ce travail.

Afin de générer les données pour paramétrer le modèle boîte noire, des simulations sont réalisées avec une sélection rigoureuse des signaux à appliquer sur les actionneurs. Un signal de Séquence Binaire Pseudo Aléatoire (SBPA) a la capacité d'exciter un système dynamique sur une large bande de fréquence [104]. Néanmoins, cette séquence ne change pas les signaux en amplitude. Il serait aussi intéressant d'appliquer des échelons de consigne aléatoire en amplitude, c'est pour cela que les Séquences Multiniveaux Pseudo Aléatoires (SMPA) ont été conçues, permettent d'allier ces deux types de consignes (aléatoires en fréquence et en amplitude) [105].

Le prétraitement des données est réalisé pour les exploiter et en tirer les informations essentielles. Cette étape est primordiale pour ne pas entraver la bonne réalisation des étapes suivantes. Concrètement, elles sont nettoyées et mises en forme par exemple avec une normalisation. Ensuite, les données sont sauvegardées dans un fichier texte ou avec un outil de gestion de base de données.

2. Il n'est pas spécifié dans cette section l'algorithme qui permet la mise en forme du code et l'exécution. Pour obtenir une idée du fonctionnement de la compilation du code Modelica et l'exécution ainsi que les difficultés de résolution il est possible de consulter les documents [99]-[103]

2.4 Paramétrisation du modèle boîte noire

2.4.1 Prédiction des états du système

Le système dynamique est considéré à temps discret et invariant dans le temps, sa notation est :

$$\bar{x}[k+1] = f(\bar{x}[k], \bar{u}[k]) \tag{2.1}$$

avec $\bar{x}$ les états du système, $\bar{u}$ les entrées du système et $k \in \mathbb{N}_{\geq 0}$ est l'indice d'échantillonnage, $\bar{x}[k] = \bar{x}(t_k)$, $t_k = k \times T_s$, avec T_s la période d'échantillonnage en seconde. Le modèle boîte noire est employé pour réaliser les prédictions des états du système au prochain échantillon $k+1$, tel que :

$$\hat{x}[k+1] = g_{nn}(\bar{x}[k], \bar{u}[k]) \tag{2.2}$$

avec $\hat{x}[k+1]$ le vecteur d'état prédit par le modèle boîte noire. $\bar{x}$ est le vecteur d'état provenant des données et $\bar{u}$ le vecteur d'entrée de commande provenant des données. g_{nn} est le modèle à paramétrer, dans ce travail de thèse, le modèle est un réseau de neurones artificiels. Par ailleurs, les échantillons doivent être mis en forme pour correspondre à l'entrée du réseau de neurones (les états et les entrées à l'instant k) et le vecteur de sortie (la prédiction des états à l'instant $k+1$).

2.4.2 Modèle boîte noire

Dans le cadre de la méthodologie, nous avons choisi de réaliser la prédiction des états du système dynamique par un réseau de neurones artificiels. Ce choix a été motivé par des études comme celles présentées dans l'article [66] ou [106] montrant que l'utilisation de réseau de neurones doit permettre de meilleurs prédictions que celles obtenues avec des modèles paramétriques. Trois types d'architectures neuronales sont sélectionnées : le perceptron multicouche, le ResNet et le PolyNet. Le perceptron est sélectionné pour sa simplicité de mise en oeuvre et il sera notre point de comparaison. Ensuite, les deux autres architectures sont dérivées des méthodes d'intégration. Les unités élémentaires sélectionnées sont ResNet pour obtenir une approche d'Euler explicite et PolyNet pour obtenir une approche d'Euler implicite. Néanmoins, d'autres réseaux ont aussi été regardés dans ce travail, voire l'annexe E.

Neurone artificiel

Le neurone artificiel est dérivé du neurone biologique. Celui-ci est complexe (cf. illustration sur la Fig.2.2 ou la description proposée dans [107]), et finalemet sa modélisation mathématique est réalisée selon plusieurs hypothèses simplificatrices, ne retenant que le principe de base de sa capacité de communication par des signaux électriques.

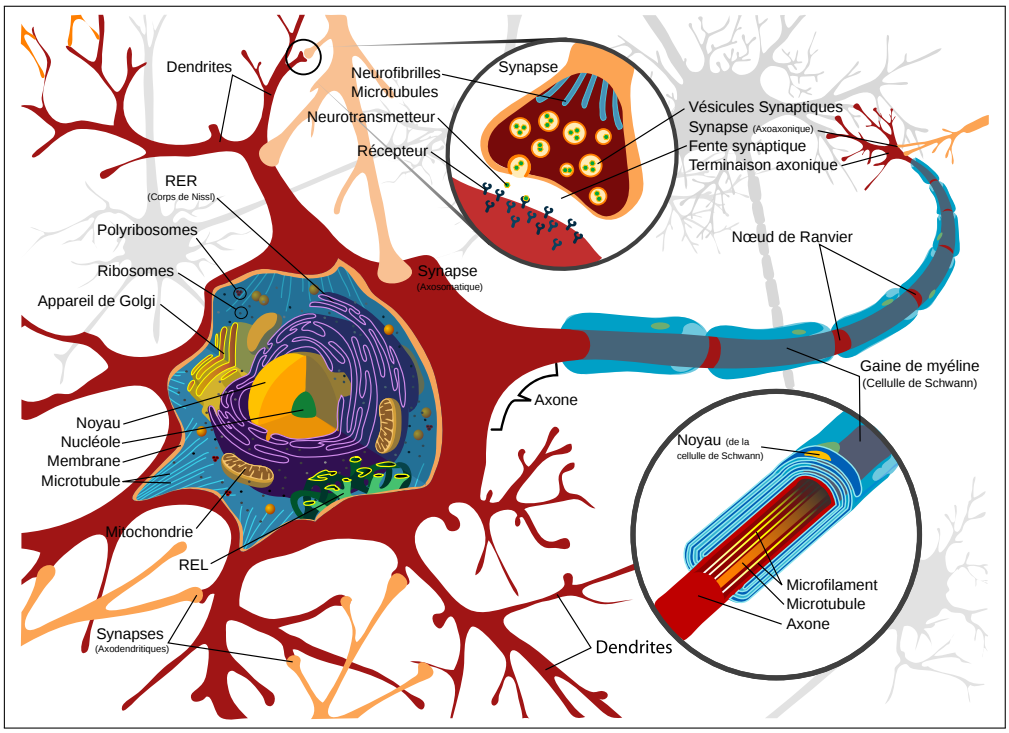

FIGURE 2.2 – Schématisation de la simplification du neurone biologique, extrait de [108].

Le premier modèle mathématique d'un neurone a été proposé par W. McCulloch et W. Pitts [109] après observation des neurones de la grenouille [110]. Le modèle possède une activation logique avec une sortie égale à 1 ou 0. Si les entrées connectées au neurone sont activées à partir d'un certain seuil, la sortie du neurone devient égale à 1 [111]. Les neurones artificiels communément utilisés actuellement pour l'apprentissage automatique sont différents du modèle de McCulloch et Pitts. Le neurone artificiel possède une fonction d'activation, des poids et un seuil [63], une illustration est visible à la Fig.2.3. Par ailleurs, sur la Fig.2.3 il est indiqué l'analogie avec le neurone biologique avec axone, synapse, dendrite, noyau et axone de sortie.

La formulation mathématique d'un neurone artificiel avec plusieurs entrées et plusieurs

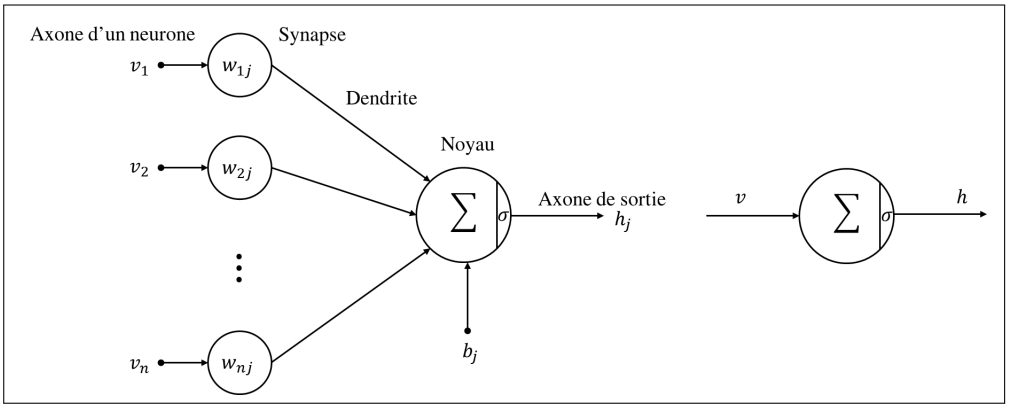

FIGURE 2.3 – Schématisation du neurone artificiel, adapté de [112] avec à gauche la représentation produit par élément et à droite la représentation vectorielle.

sorties en représentation produit par élément est égale à [113] :

$$h_j = \sigma(b_j + \sum_j w_{ij} v_j) \tag{2.3}$$

avec h_j la sortie, b_j le seuil, w_{ij} le poids de connexion de l'unité i à l'unité j et v_j la j-ième entrée. La représentation mathématique d'un neurone artificiel avec plusieurs entrées et plusieurs sorties en représentation matricielle est privilégiée dans ce travail [113] :

$$h = \sigma(W v + b) \tag{2.4}$$

avec h le vecteur de sortie du neurone artificiel, v le vecteur d'entrée du neurone artificiel, W la matrice de poids et b le vecteur contenant les seuils. Par ailleurs, σ est la fonction d'activation. Il existe un certain nombre de fonctions d'activation pour le neurone artificiel. Nous pouvons citer par exemple les fonctions sigmoïde ou tangente hyperboliques [114]. Les poids W sont à régler afin que le vecteur de sortie h coïncide avec le signal à atteindre pour une entrée v considérée ; le réglage de ces points est réalisé durant la phase dite "d'apprentissage". Généralement, un neurone n'est pas suffisant et ne contient pas assez de paramètres pour une bonne prédiction. Ainsi les neurones artificiels sont utilisés avec un empilement horizontal ou vertical afin de réaliser des réseaux profonds [60]. Les trois réseaux sélectionnés dans ce travail sont présentés dans les sections suivantes (perceptron multicouche en anglais *multilayer perceptron* ou bien aussi *Feedforward Neural Network* (Fnn), Residual layer Network (ResNet) et Polyinceptions Networks (PolyNet)).

Perceptron multicouches

Les neurones artificiels sont empilés en profondeur pour former le perceptron multicouche. Une représentation du perceptron multicouche avec quatre couches cachées est visible sur la Fig.2.4. Dans le cas particulier où le réseau possède une couche cachée, un nombre fini de neurones et la fonction d'activation sigmoïde, le réseau permet d'approximer toute fonction continue non linéaire [115], [116]. Ainsi, le perceptron multicouche s'avère être un approximateur universel [117]. L'équation (2.5) donne la formulation mathématique générique d'un perceptron multi-couches [118] :

$$h = \mathcal{G}(v) = g_{h^1}(v) \circ g_{h^2}(h^1) \circ \ldots \circ g_{h^{i-1}}(h^{i-2}) \circ g_{h^i}(h^{i-1}) \tag{2.5}$$

avec h le vecteur de sortie du réseau de neurones, v le vecteur d'entrée du réseau et $\circ$ est le produite de Hadamard. g_{h^i} est la i-ème couche cachée du réseau avec : $h^i = g_{h^i}(h^{i-1}) = \sigma^i(W^i h^{i-1} + b^i)$ avec h^i la sortie, h^{i-1} l'entrée, σ^i la fonction d'activation, b^i le seuil de la couche cachée i et W^i la matrice de poids.

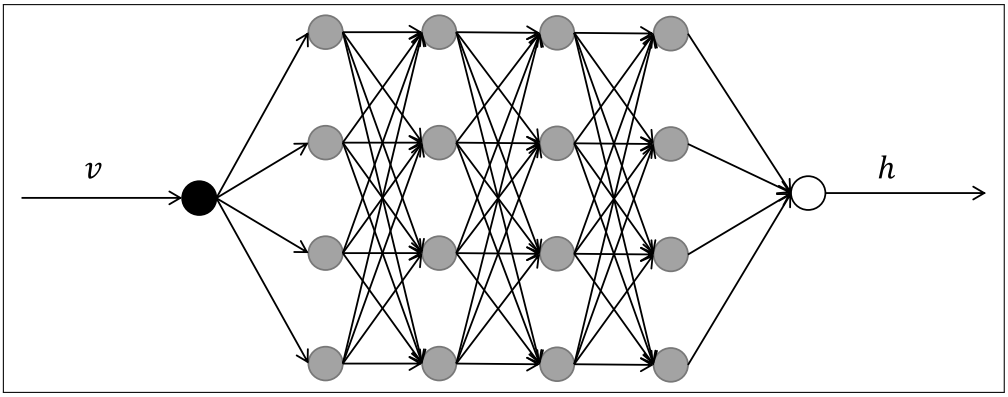

FIGURE 2.4 – Schématisation du perceptron multi-couches profond, avec une entrée, une sortie et quatre couches cachées.

Residual neural network (ResNet)

L'architecutre *residual neural network* a été proposée pour la première fois en 2016 pour la reconnaissance d'image [119] [120]. Depuis, de nombreux travaux de recherches ont employé ce réseau, par exemple l'automate jouant au jeu de go : *AlphaGo Zero* [121] ou la traduction automatique du langage [122]. Le réseau est aussi employé pour la prédiction de systèmes dynamiques [123]. En effet, il a été observé pour la première fois dans le travail de Weinan

le lien entre la méthode d'intégration de Euler explicite et l'architecture ResNet [124]. La méthode d'Euler explicite pour l'intégration d'une équation différentielle continue notée par $\dot{y}(t) = g(y(t))$, $y(0) = y_0$ est [125] :

$$y[k+1] = y[k] + \Delta_k g(y[k]) \tag{2.6}$$

avec $y[k+1]$ la solution à $k+1$ de l'intégration de l'EDO et Δ_k la période d'échantillonnage. Nous pouvons maintenant présenter la formulation mathématique d'une unité du réseau ResNet [120] :

$$h = v + \mathcal{G}(v) = (I + \mathcal{G})(v) \tag{2.7}$$

avec h la solution en sortie de l'unité, $\mathcal{G}$ la transformation non linéaire neuronale et v l'entrée de la couche ResNet. Dans le cadre de ce travail, $\mathcal{G}$ est issue de l'Eq.2.5. Une illustration d'une cellule ResNet est visible à la Fig.2.5.

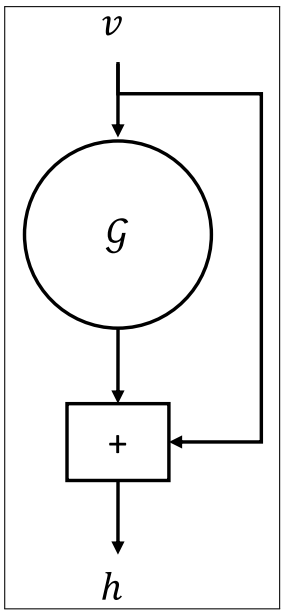

FIGURE 2.5 – Schématisation d'une unité de ResNet [119].

ResNet permet de définir des réseaux profonds avec plusieurs centaines de couches tout en

maintenant l'entrainement réalisable. On modélise un ResNet à n couches comme suit :

$$h^1 = (I + \mathcal{G})(v) \tag{2.8}$$

$$h^2 = (I + \mathcal{G})(h^1) = (I + \mathcal{G})^2(v) \tag{2.9}$$

$$\vdots$$

$$h^n = (I + \mathcal{G})^n(v) \tag{2.10}$$

avec h^n la sortie du réseau avec n couches et v le vecteur d'entrée. Il est maintenant aisé d'observer que l'empilement d'unité de ResNet permet de relier le ResNet avec une méthode d'intégration d'ordre plus élevée où chaque couche est une étape de l'intégration. Une illustration de deux couches ResNet est visible à la Fig.2.6. Par ailleurs, il est à noter que le réseau ResNet avec plusieurs couches possède le même réseau $\mathcal{G}$ à chaque couche. Cet élément est différent de la présentation du ResNet originale ou chaque couche possède ses neurones avec ses propres poids [119]. La conservation du même réseau est un hypothèse dans ce travail afin de comparer les performances de ResNet par rapport au Fnn et pour faire le lien avec la méthode d'intégration d'ordre élevé.

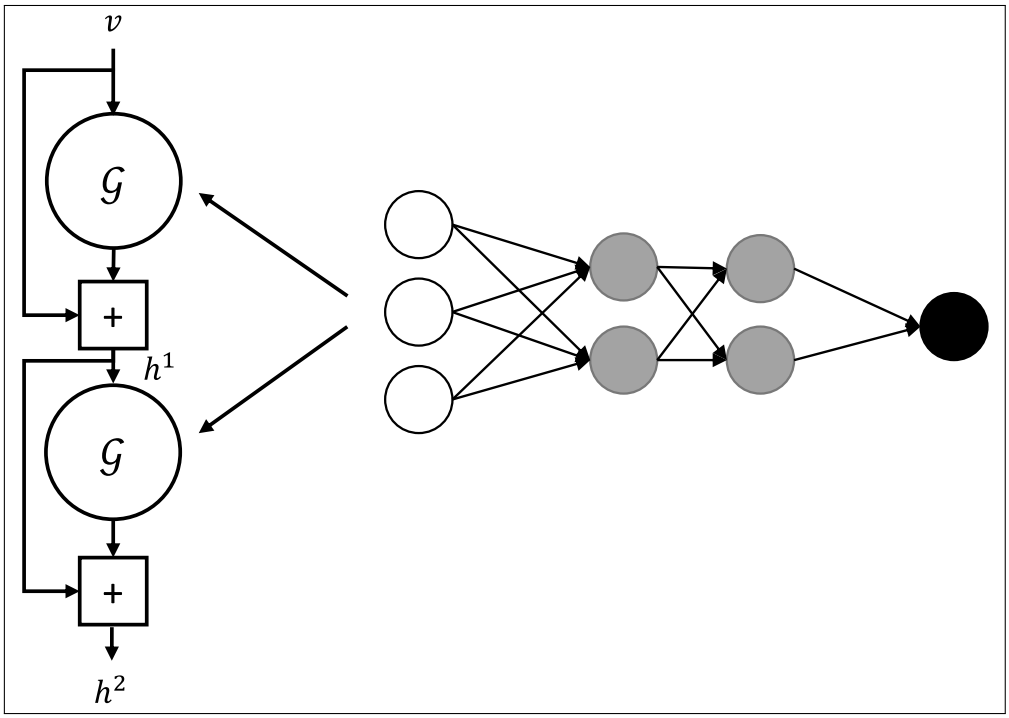

FIGURE 2.6 – Schématisation du ResNet avec deux couches.

Polyinceptions neural network (PolyNet)

L'architecture polyinceptions neural network est présentée en 2017 [126] pour la reconnaissance d'images dans le but d'améliorer les performances du ResNet. En outre, il a été observé que PolyNet est relié à la méthode d'intégration d'Euler implicite [127], ce qui motive son adoption dans ce travail. La méthode d'Euler implicite pour l'intégration d'une équation différentielle continue notée par $\dot{y}(t) = g(y(t))$, $y(0) = y_0$ est [127] :

$$y[k+1] = y[k] + \Delta_k g(y[k+1]) \tag{2.11}$$

$$y[k+1] - \Delta_k g(y[k+1]) = y[k] \tag{2.12}$$

$$y[k+1] = (I - \Delta_k g)^{-1} y[k] \tag{2.13}$$

avec $y[k+1]$ la solution à $k+1$ et Δ_k la période d'échantillonnage. La formulation mathématique de PolyNet est [126] :

$$h = v + \mathcal{G}(v) + \mathcal{G}(\mathcal{G}(v)) = (I + \mathcal{G} + \mathcal{G}^2)(v) \tag{2.14}$$

avec h la solution en sortie de la cellule PolyNet considérée, $\mathcal{G}$ la transformation neuronale et v l'entrée de la cellule PolyNet. $\mathcal{G}$ est défini par l'Eq.2.5. Une illustration de PolyNet est visible à la Fig.2.7. Le lien avec la méthode d'Euler implicite nécessite d'utiliser l'approximation : $\frac{1}{1-x} \approx \sum_{n=0}^{\infty} x^n$, menant à : $(I - \Delta_t f)^{-1} = I + \Delta_t f + (\Delta_t f)^2 + \ldots + (\Delta_t f)^n + \ldots$.

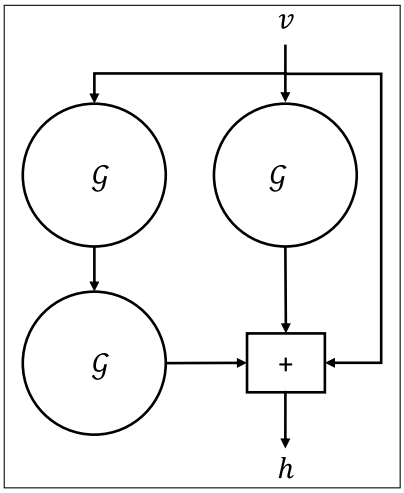

FIGURE 2.7 – Schématisation d'une unité de PolyNet [126].

PolyNet permet d'adopter des réseaux profonds avec plusieurs centaines de couches tout en maintenant l'entrainement réalisable. Une illustration du PolyNet avec deux couches est

visible à la Fig.2.8. La formulation mathématique de PolyNet avec l'augmentation du nombre de couches est :

$$h^1 = (I + \mathcal{G} + \mathcal{G}^2)(v) \tag{2.15}$$

$$h^2 = (I + \mathcal{G} + \mathcal{G}^2)h^1 = (I + \mathcal{G} + \mathcal{G}^2)^2(v) \tag{2.16}$$

$$\vdots$$

$$h^n = (I + \mathcal{G} + \mathcal{G}^2)^n(v) \tag{2.17}$$

avec h^n la sortie du réseau avec n couches, v l'entrée, et $\mathcal{G}$ la transformation non linéaire neuronale. Pour rappel $\mathcal{G}$ est défini par l'Eq.2.5. Par ailleurs, il est à noter que le réseau PolyNet avec plusieurs couches possède le même réseau $\mathcal{G}$ à chaque couche. Cet élément est différent de la présentation du PolyNet ou chaque couche possède ses neurones avec ses propres poids [126]. La conservation du même réseau est un hypothèse dans ce travail afin de comparer les performances de PolyNet par rapport au Fnn et pour faire le lien avec la méthode d'intégration d'ordre élevé.

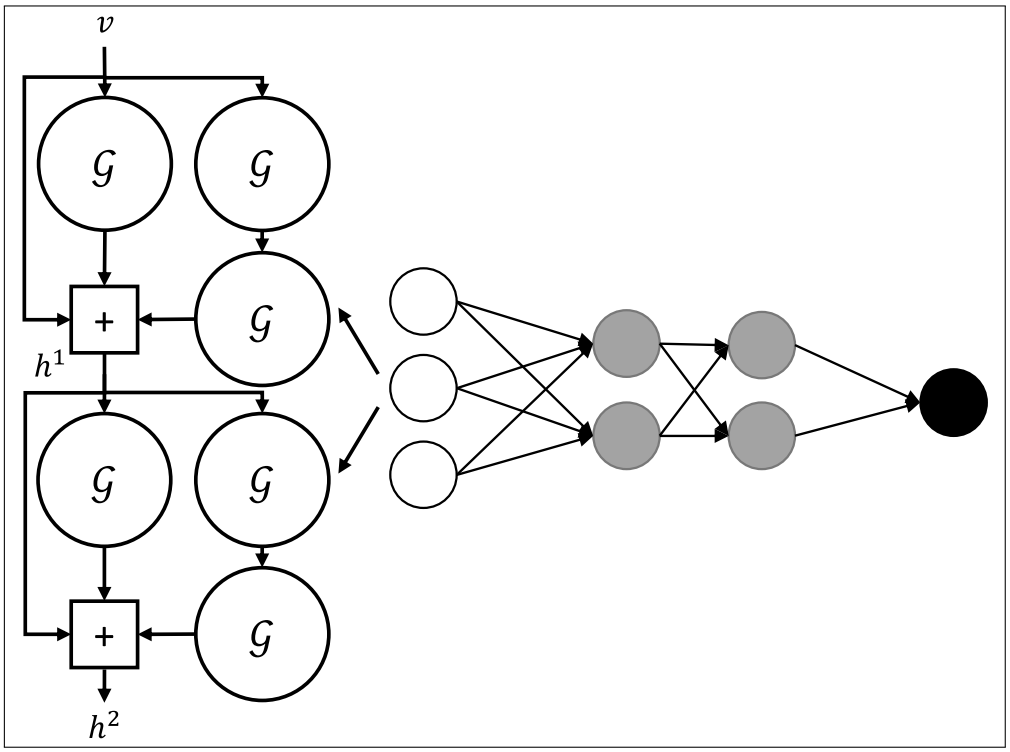

FIGURE 2.8 – Schématisation du PolyNet avec deux couches.

Lien avec la prédiction du système dynamique

Il reste maintenant une dernière étape, pour adapter le réseau PolyNet et ResNet à l'identification du système dynamique et la prédiction des états à l'instant $k+1$ comme résumé pour l'Eq.2.2. En effet, les réseaux ResNet puis PolyNet supposent de possèder la même dimension pour les vecteurs d'entrée et de sortie. Or concrètement les systèmes dynamiques ont des tailles de vecteur d'entrée et de sortie différentes, avec les signaux (x, u) en entrée, et x en sortie. Pour adapter ces deux types de réseau à notre problème, des couches d'entrée et de sortie sont ajoutées, Fig.2.9. Néanmoins, ces couches sont construites avec des fonctions d'activation identité pour ne pas réaliser de transformation non-linéaire des données. Les couches d'entrée et de sortie sont définies par :

$$h^{in} = W^{in}(\bar{x}[k], \bar{u}[k]) + b^{in} \tag{2.18}$$

$$\vdots$$

$$\hat{x}[k+1] = W^{out}(h^n) + b^{out} \tag{2.19}$$

avec h_{in} la sortie de la couche d'entrée, W^{in} les poids de la couche d'entrée, b^{in} les seuils de la couche d'entrée, $\bar{x}[k]$ l'état du système, $\bar{u}[k]$ les entrées de commande, $\hat{x}[k+1]$ les prédictions des états, W^{out} les poids de la couche de sortie, b^{out} les seuils de la couche de sortie, h^n la sortie de la couche n provenant de ResNet ou PolyNet.

2.4.3 Entraînement et réglage des hyperparamètres

L'entraînement du réseau de neurones consiste à paramétrer ses poids et seuils pour atteindre la sortie recherchée du modèle. L'entrainement est réalisé à l'aide de l'algorithme dominant de rétropropagation du gradient [128], [129] dont le but est de minimiser une fonction coût notée par g_{loss}. Elle est liée à la mesure de la performance du modèle boîte noire et elle sera précisée dans la section suivante 2.4.4.

Les hyperparamètres regroupent les paramètres du réseau de neurones et de l'algorithme du gradient. Les hyperparamètres considérés sont : le nombre de neurones, le nombre de couches, la fonction d'activation, la taille du batch, le nombre d'époques, l'optimiseur utilisé et ses paramètres (taux d'apprentissage et les momentums [130]). Ils ont une influence importante sur la durée d'apprentissage du réseau ainsi que la performance finale du modèle, un bon réglage de ceux-ci est d'une importance capitale [131]. Obtenir de bons hyperparamètres avant l'apprentissage est délicat voire impossible, de même que tester toutes les possibilités. Une mé-

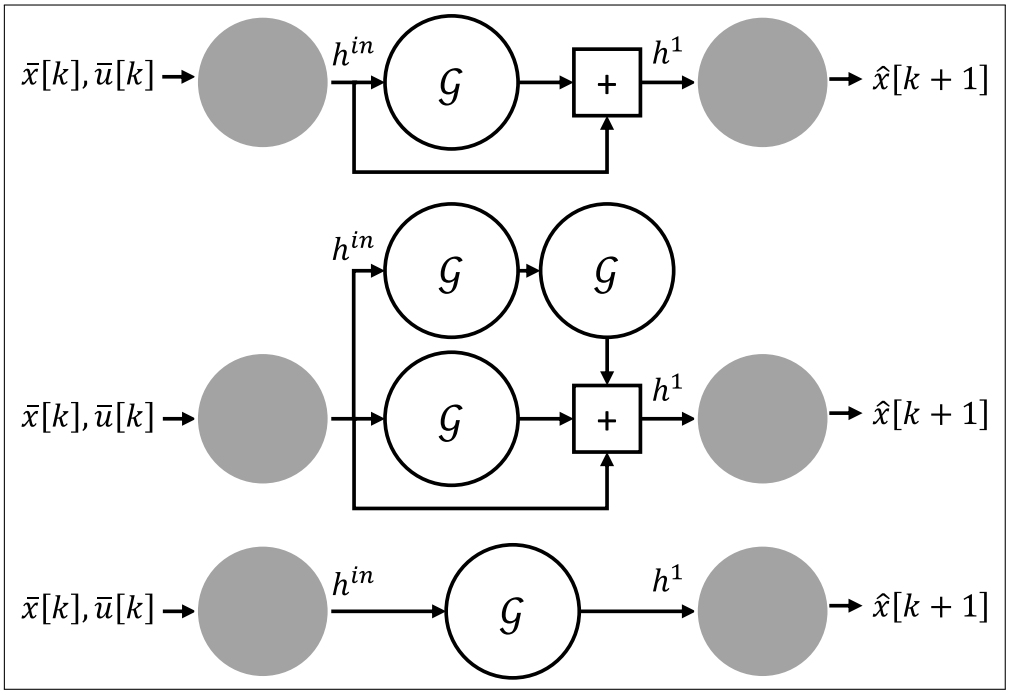

FIGURE 2.9 – Schématisation du Resnet (haut), PolyNet (milieu) et Fnn (bas) avec les couches d'entrée et de sortie. (neurones en gris).

thode exploitée dans ce travail est de réaliser une optimisation pour régler les hyperparamètres. La fonction coût à minimiser est la même que lors de l'entrainement, voir section 2.4.4. Une illustration est visible à la Fig.2.10, les différentes étapes de l'algorithme sont :

- Une sélection des hyperparamètres est réalisée ;
- Un réseau de neurones avec les hyperparamètres obtenus à l'étape précédente est formé ;
- Le réseau est entraîné ;
- Lorsque que l'entraînement est terminé, la performance du réseau est évaluée avec un jeu de données différent de celui de l'entraînement par la fonction de coût en fonction des hyperparamètres sélectionnés ;

Au démarrage, les hyperparamètres sont choisis arbitrairement. Puis à mesure des itérations, leurs influences sur la performance de prédiction du modèle boîte noire est pris en compte sur la performance de prédiction du modèle boîte noire.

Le problème d'optimisation avec les hyperparamètres du réseau de neurones est non-convexe et non dérivable [131]. L'utilisation d'algorithme métaheuristique est requise. Plusieurs solutions existent telles que l'algorithme de colonie de fourmis ou l'algorithme génétique [132]. Une grande partie des algorithmes métaheuristiques ont un point en commun : le vivant a été observé puis son fonctionnement transposé en algorithme d'optimisation [109]. De

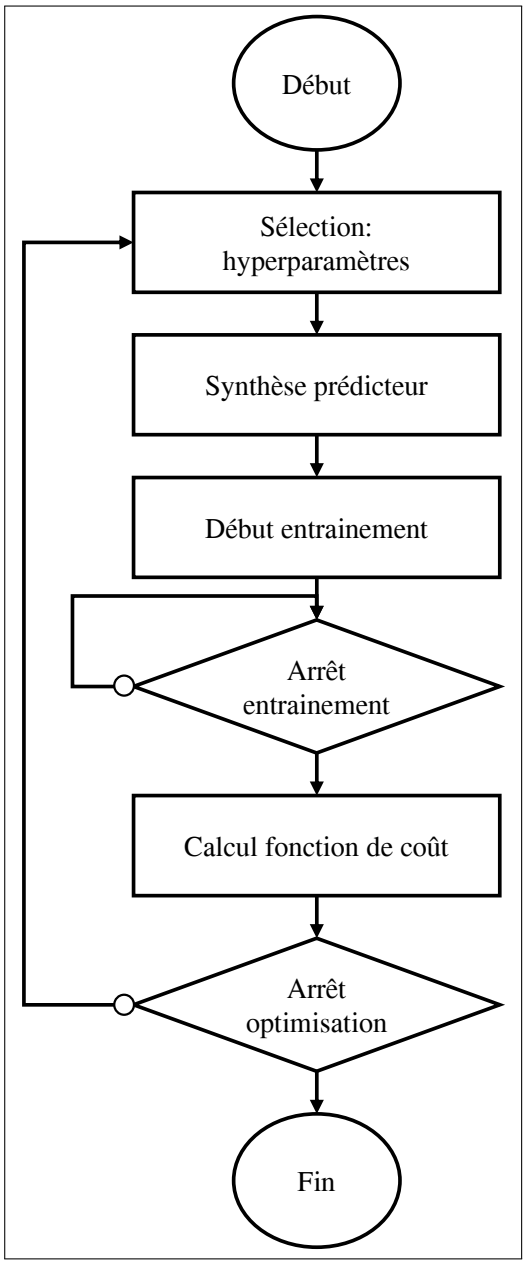

FIGURE 2.10 – Algorithme optimisation des hyperparamètres du réseau de neurones (prédicteur des états).

plus, ces algorithmes itératifs peuvent présenter un temps de calcul et de convergence longs. L'implémentation de certains algorithmes métaheuristiques est disponible dans des outils informatiques tels que Deap [133] avec le langage Python ou bien BlackBoxOptim.jl [134] avec le langage Julia. La formulation mathématique du problème d'optimisation est :

$$\min_{H} \quad g_{loss} \tag{2.20a}$$

$$\text{sous contraintes} \quad g_{nn}(H) \tag{2.20b}$$

$$H \in \mathcal{H} \tag{2.20c}$$

avec g_{loss} est la fonction de coût à minimiser de l'Eq.2.20a avec les données considérées pour l'entrainement. La fonction de coût peut être vue comme la mesure de la fidélité du réseau. Eq.2.20b est la contrainte du réseau de neurones avec la réalisation de l'entrainement. Eq.2.20c présente les contraintes des hyperparamètres admissibles.

2.4.4 Mesure de la fidélité

Il est nécessaire de posséder une notion quantitative pour mesurer la performance des réseaux de neurones à reproduire le signal de sortie. Elles représente la similarité entre les données source à disposition et les sorties du modèle à paramétrer [135]. La fonction utilisée dans ce travail de recherche est l'Erreur Quadratique Moyenne, *Mean Square Error* (MSE) [135] :

$$g_{loss} = \frac{1}{N_b} \sum_{n=1}^{N_b} (\bar{x}_n - \hat{x}_n)^2 \tag{2.21}$$

avec N_b le nombre total d'échantillons, $\hat{x}$ la sortie du modèle g_{nn} et $\bar{x}_n$ les échantillons cibles. Par ailleurs, d'autres fonctions sont visibles à l'annexe D. Pour un même volume de données à disposition, plus le taux de mesure de la fidélité est bas, de meilleur niveau est la prédiction, il cherchera donc à minimiser g_{loss}.

2.4.5 Implémentation numérique

Les réseaux de neurones artificiels sont implémentés sur une machine de calcul à l'aide d'un outil informatique dédié pour réaliser la formulation des réseaux, l'entrainement et la validation. Il existe plusieurs outils à disposition. Par exemple, nous pouvons indiquer Flux [136] ou Knet [137] avec langage julia, Machine Learning Toolbox avec Matlab [138], PyTorch avec le langage Python [139] ou Keras avec le langage Python [140]. Ces outils permettent de réaliser des

prétraitements sur les données, de construire des architectures de réseau de neurones, de réaliser l'entrainement et l'optimisation à base de l'algorithme de rétropropagation du gradient [128], [129]. Par ailleurs, tous ces outils autorisent l'utilisation de matériel informatique spécifique pour accélérer l'apprentissage tel que les GPU [141] ou les *Tensor Processing Unit* (TPU) [142].

2.5 Synthèse du régulateur

2.5.1 Commande prédictive

La commande prédictive ou *Model Predictive Control (MPC)* en anglais est une technique de commande avancée, qui permet notamment de piloter des systèmes multivariables (i.e. présentant plusieurs entrées de commande et sorties de mesure). Sa spécificité réside dans le fait qu'elle exploite un modèle interne du processus à piloter, lui permettant de prédire sur un certain horizon temporel son comportement, et donc d'optimiser sur cette base ses signaux de commande. L'implémentation d'une commande MPC implique donc la résolution en temps réel d'un problème d'optimisation sur la base d'un critère définissant les objectifs de commande, typiquement un suivi de consigne, cf. Fig.2.11. On attribue la création de cette technique à J. Richalet, en 1978 [39].

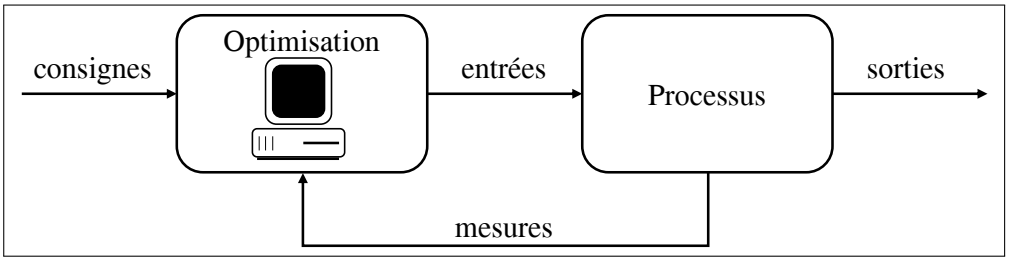

FIGURE 2.11 – Schématisation d'une boucle de contrôle avec la MPC.

Formulation

Afin de poursuivre la méthodologie et de se rapprocher de l'implémentation informatique, la formulation de la MPC utilise le modèle boîte noire de l'équation (2.2) pour la prédiction du système dynamique. Tout d'abord, donnons les éléments de notation suivants :

- Le vecteur des états prédits du système sont notés $\hat{x}$.
- Le vecteur des entrées de commande sont notés u.
- Le vecteur des consignes des états sont notés x^r.

— Le vecteur des consignes des entrées de commande sont notés u^r.
— L'écart entre les états du système et la consigne associée est noté par $\tilde{x} = \hat{x} - x^r$.
— L'écart entre les entrées de commande et la consigne associée est notée par $\tilde{u} = u - u^r$.
— L'indice du temps de l'échantillon discret est $i \in \mathbb{N}_{\geq 0}$
— La modélisation du système dynamique par un prédicteur neuronal est notée par g_{nn}.
— La fonction de coût à minimiser est notée par $\mathcal{J}$.

Nous pouvons maintenant spécifier le problème d'optimisation de la MPC [143] :

$$\min_{\tilde{u}} \mathcal{J} \tag{2.22}$$

$$\text{s.c. } \hat{x}_{i+1} = g_{nn}(\hat{x}_i, u_i) \tag{2.23}$$

$$\hat{x}_i \in \mathcal{X} \tag{2.24}$$

$$u_i \in \mathcal{U} \tag{2.25}$$

$$\hat{x}_0 = \bar{x}(t) \tag{2.26}$$

$$\hat{x}_N \in \mathcal{X}_f \tag{2.27}$$

$$\tilde{x}_i = \hat{x}_i - x_i^r \tag{2.28}$$

$$\tilde{u}_i = u_i - u_i^r \tag{2.29}$$

avec l'Eq. 2.22 la fonction coût à minimiser, $\tilde{u}$ est la séquence de commande d'entrée aux écarts : $\tilde{u} := \tilde{u}_0, \dots, \tilde{u}_{N-1}$, N l'horizon, l'Eq. 2.23 la prédiction des états du système dynamique, l'Eq. 2.24 les contraintes sur les états, l' Eq. 2.25 les contraintes sur les entrées de commande, l'Eq. 2.26 l'initialisation des états avec les mesures du système, l'Eq.2.27 la contrainte sur les états finaux, l'Eq.2.28 l'écart entre les états du système et la consigne associée, l'Eq.2.29 l'écart entre les entrées de commande et la consigne associée.

L'entrée de commande appliquée aux actionneurs est le premier vecteur de pilotage calculé par le problème d'optimisation : u_0^*. Ensuite comme illustrée par la Fig.2.12, au pas de temps suivant on réinitialise le problème d'optimisation que l'on résout de nouveau, selon le principe d'un horizon glissant. Cela constitue alors une boucle fermée "implicite".

L'algorithme 2.13 présente les étapes de pilotage de la commande MPC, qui sont : mesurer les états du système, résoudre le problème d'optimisation de la MPC, appliquer la commande puis attendre la prochaine itération de la MPC.

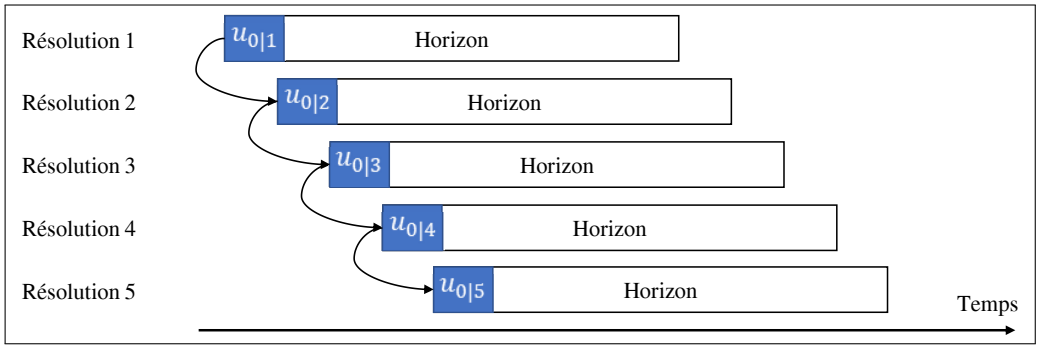

FIGURE 2.12 – Horizon glissant occasionné par la résolution itérative de la commande MPC.

Fonction coût

Un défi important de la MPC est d'assurer à chaque exécution la stabilité et la faisabilité. Dans sa formulation originelle le critère à optimiser est issue du critère type "Linéaire Quadratique" (LQ), cherchant le compromis entre la minimisation du suvi de trajectoire pour l'état et la commande. La difficulté vient du fait que l'optimisation doit se faire ici sur un horizon fini, qui amène au défi d'assurer à chaque exécution la stabilité et la faisabilité. Cette question est de nos jours bien adressée dans la littérature [144]. Une solution couramment usitée est d'ajouter un coût final dans le critère, ainsi qu'une contrainte finale. Finalement, la fonction de coût de la MPC, noté par $\mathcal{J}$ peut être défini ainsi :

$$\mathcal{J} = \tilde{x}_N^T P \tilde{x}_N + \sum_{i=0}^{N-1} \tilde{x}_i^T Q \tilde{x}_i + \tilde{u}_i^T R \tilde{u}_i \tag{2.30}$$

avec P, Q et R les matrices de pondération, sélectionnées comme matrices semi-définies positives [145] et de taille appropriée. $\tilde{u}$ est la séquence aux écarts de la commande d'entrée : $\tilde{u} := \tilde{u}_0, \dots, \tilde{u}_{N-1}$, N l'horizon, N est l'horizon, ^{T} est la transposée de la matrice et i est l'instant d'échantillonnage.

Le coût final : $\tilde{x}_N^T P \tilde{x}_N$ est associé à une fonction de Lyapunov pour assurer la stabilité asymptotique en boucle fermée par la commande MPC [143], [146]. Pour un système linéaire stable (ou un système non linéaire, linéarisé et stabilisé à l'équilibre), P peut-être calculé par la solution de l'équation discrète de Riccati lors de la synthèse d'un régulateur LQR sans contrainte à horizon infini [143], [144], [147] :

$$P_\infty = A^T P_\infty A + Q - A^T P_\infty B (B^T P_\infty B + R)^{-1} B^T P_\infty A \tag{2.31}$$

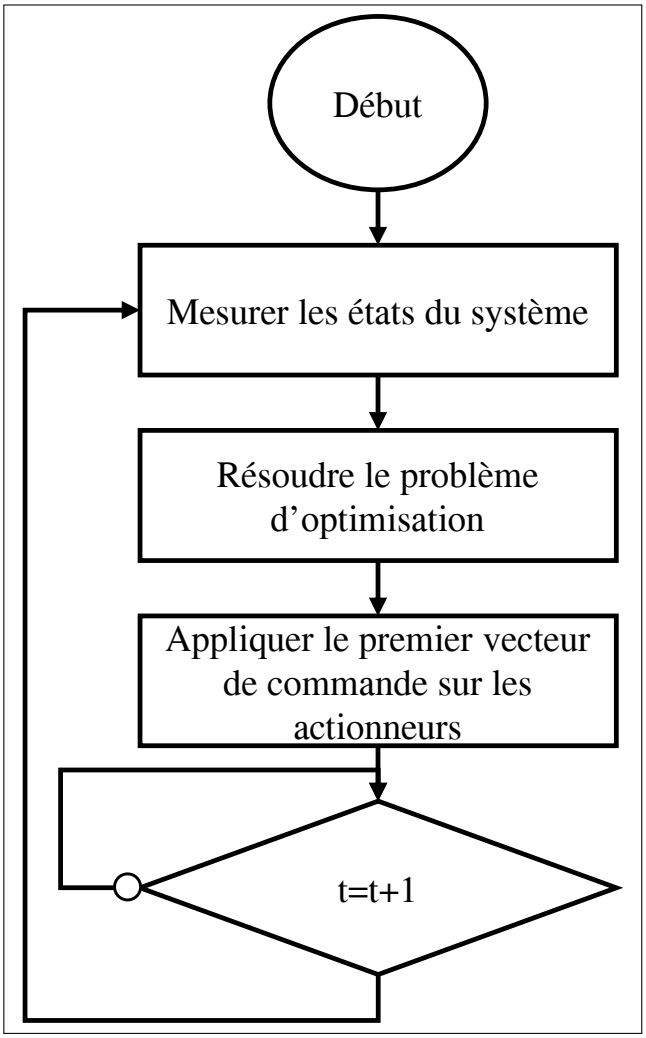

FIGURE 2.13 – Algorithme de la commande MPC.

avec, A, B les matrices du système linéaire, Q, R les matrices de poids de la fonction coût, enfin P_∞ la solution de l'équation discrète de Riccati. P peut aussi être calculée par la résolution de l'équation discrète de Lyapunov [144], [148] :

$$P - A^T P A = Q \tag{2.32}$$

avec,A la matrice des états et Q la matrice de poids de la fonction coût. La résolution de ces équations (2.31) ou (2.32) s'accompagne toujours de la contrainte P>=0 (P semi-définie positive), la solution pourra être obtenue avec un solveur adapté, par exemple SCS [149].

Contrainte finale

La contrainte finale est associée à la faisabilité récursive. Celle-ci implique une faisabilité pour les prochains états et entrées sous contraintes ainsi que pour tous les autres calculés grâce à la MPC. Si la faisabilité récursive n'est pas atteinte, il n'y a pas de garantie que les contraintes soient respectées [150].

Il est possible tout en assurant la faisabilité et la stabilité de renseigner une contrainte finale égale à zéros dans le problème d'optimisation [146], [151] :

$$\tilde{x}_N = 0 \tag{2.33}$$

avec $\tilde{x}_N$ les états finaux à N. Ce choix est simple, ne nécessite pas de traitement hors-ligne de la MPC, néanmoins il peut poser des difficultés de calcul pour un horizon N réduit.

La seconde possibilité avec l'assurance de la faisabilité et la stabilité, est d'imposer une contrainte finale de contraction ; la MPC est appelée une MPC contractive [146], [148], [152]. La contrainte finale est remplacée par une contraction selon une norme sur les états finaux par rapport aux états initiaux [146], [152] :

$$M(\tilde{x}_N) \leq \alpha M(\tilde{x}_0) \tag{2.34}$$

avec $\tilde{x}_0$ les états à l'initialisation de la MPC (mesure du système), α le coefficient de contraction $[0, 1[$, $\tilde{x}_N$ les états finaux et M est une matrice définie positive. La MPC contractive ne nécessite pas de traitement hors-ligne, elle offre plus de latitude pour l'optimisation. Néanmoins, de nouveaux paramètres sont nécessaires, une contrainte quadratique est ajoutée dans le problème d'optimisation, et elle peut poser des difficultés pour la convexité.

La troisième possibilité avec l'assurance de la faisabilité et la stabilité est l'utilisation d'une

région terminale. La région offre plus de latitude à l'optimiseur, néanmoins il est nécessaire de réaliser un traitement hors ligne pour calculer cette région terminale. Dans le cadre d'un système linéaire, il est possible de sélectionner $\mathcal{X}_f$ ensemble invariant de la boucle fermée $x_{i+1} = Ax_i + BF_\infty x_i$, régulé par le régulateur LQR F_∞ [143], [146]. Afin de calculer cet ensemble pour le système dynamique linéaire, des méthodes sont décrites dans le chapitre 10 de [153], le chapitre 2 du livre [154], ou bien avec le calcul des ellipsoïdes dans [150]. Dans le cadre d'une MPC non linéaire, le calcul de l'ensemble invariant est plus difficile à obtenir [155]. Il est envisageable de linéariser le système non linéaire puis de calculer la région terminale [147], [156].

2.5.2 Implémentation numérique

La MPC est programmée en langage informatique dédié à la modélisation de problème d'optimisation. Il existe un certain nombre d'outils à disposition, des outils commerciaux et avec licence open source. Nous pouvons lister : *A Mathematical Programming Language* (AMPL) [157], *General Algebraic Modeling System* (Gams) [158], *Optimization toolbox* de Matlab [159], CasADi [160], Yalmip [161], Pyomo avec le langage Python [162], OR-tools de Google [163] et JuMP [164] avec le langage julia [165]. Dans le cadre de ce travail de recherche, nous avons sélectionné l'outil JuMP pour la rapidité d'exécution des programmes julia et de la licence open source.

Deux méthodes d'implémentation de la MPC sont possibles selon la sélection des variables de décision [3]. La première méthode est appelée *single shooting* et la seconde *multiple shooting* [166]. La méthode *single shooting* autorise comme variables de décision les entrées de commande de la MPC alors que la méthode *multiple shooting* utilise comme variables de décision les entrées de commande et les états. L'une et l'autre des méthodes possèdent des avantages et des inconvénients [144], [167]. La méthode *single shooting* permet la réduction du problème d'optimisation et la méthode *multiple shooting* permet une simplification de la formulation du problème d'optimisation, néanmoins il y a plus de variables de décision.

Classiquement, la MPC est classifiée comme un problème d'optimisation quadratique linéaire ou non linéaire sous contraintes sans variable entière à manipuler. Des solveurs de résolution sont adaptés pour ces types de problèmes. Par exemple, lorsque la MPC est linéaire il existe *Object-Oriented Software for Quadratic Programming* OOQP [168], *Operator Splitting Solver for Quadratic Programs* OSQP [169] ou *Conic operator splitting method* COSMO [170]. Lorsque

3. Variable de décision : variable du problème d'optimisation à optimiser.

la MPC est non-linéaire il existe *Interior Point OPTimizer* Ipopt [171] ou Knitro [172]. Les solveurs implémentent un ou plusieurs algorithmes de résolution. Une approche maintenant classique de résolution est l'emploi de la méthode de point intérieur pour résoudre les problèmes linéaires [173] et non linéaires [174]. La méthode de point intérieur est implémentée avec les solveurs OOQP, Ipopt et Knitro. Il existe aussi l'algorithme *Alternating Direction Method of Multipliers* (ADMM) [175] pour la résolution des problèmes linéaires, il est implémenté par les solveurs OSQP et COSMO. Par ailleurs, il est à noter que l'algorithme ADMM est parallélisable, le solveur OSQP a été implémenté dans le but d'utiliser la capacité de parallélisation sur GPU [176]. Les GPU ne sont pas les seuls circuits spécialisés permettant la résolution de la MPC. Il est possible de réaliser des implémentations avec les circuits logiques programmables, *Field-programmable gate array* (FPGA) [177]. Finalement, dans ce travail nous considérons les solveurs de résolution non-linéaire Ipopt et Knitro, car la MPC synthétisée sur la base d'un modèle boîte noire réseau de neurones est non-linéaire.

2.6 Evaluation du pilotage

Une fois la loi de commande prédictive basée sur le modèle boîte noire conçue, elle peut être testée avant implémentation sur le système final sur le modèle fin multiphysique pour validation et évaluation des performances du pilotage. Les indices de performance à considérer sont de plusieurs ordres. Premièrement, il est intéressant de comparer l'algorithme de pilotage avec des régulateurs classiques employés dans l'industrie tels que des régulateurs PID ou des régulateurs à base de règles expertes. Deuxièmement, le respect de chaque contrainte sur les états et les entrées de commande est évalué. Troisièment, il est regardé le temps de calcul de la MPC afin de vérifier qu'il ne soit pas supérieur au temps d'échantillonnage. Finalement, les erreurs de consigne sont regardées pour évaluer la capacité du régulateur à amener les états aux consignes.

2.7 Conclusion

Dans ce chapitre une méthodologie de synthèse d'une commande prédictive basée sur un modèle interne est présentée. Cette méthodologie est basée finalement sur deux étapes successives de modélisation, tout d'abord une modélisation fine multiphysique du système réel à piloter, puis la conception d'un modèle de type boîte noire, renseigné sur la base des signaux issus du modèle fin, modèle qui servira de noyau fondamental à la loi de commande.

Il est à noter que cette méthodologie peut-être itérative, afin de tester plusieurs commandes prédictives et plusieurs jeux de données. Par ailleurs, afin d'accorder le modèle multiphysique de simulation au système visé, deux chemins sont possibles. La première méthode consiste à utiliser les paramètres du système réel, par exemple pour un système énergétique les puissances nominales ou pour des stockages l'énergie stockable. La seconde méthode utilise les données mesurées du système réel ; le modèle multiphysique est "callé" avec une optimisation des paramètres. Dans le cas où les données de mesures sont à disposition en nombre suffisant, il peut être envisagé d'utiliser ces données directement pour identifier le modèle boîte noire. En outre, étant dans le cadre de simulation numérique, plusieurs combinaisons de modèle boîte noire peuvent être envisagées pour comparaison. Dans ce travail, nous avons privilégié la première méthode par l'utilisation des paramètres généraux du système réel pour simplifier le développement de l'algorithme.

Chapitre 3

Validation de la méthodologie

Résumé

Ce chapitre expose la validation de la méthodologie présentée dans le chapitre 2 pour synthétiser une commande prédictive dédié au pilotage d'un système dynamique. Cette méthodologie est validée grâce à un *benchmark* utilisé par les automaticiens et des simulations numériques. Les résultats montrent que la démarche exposée est en mesure de simplifier la synthèse du régulateur, notamment la conception de son modèle interne, tout en gardant un niveau de précision et donc de performance accrus.

Une présentation de *benchmarks* d'automatique est visible à la section 3.1, l'application de la méthodologie est réalisée à la section 3.2. Ensuite, une discussion des résultats et de l'application de la méthodologie est présentée à la section 3.3, puis la section 3.4 conclut le chapitre.

3.1 *Benchmarks* d'automatique

En automatique il existe des systèmes dynamiques qui sont considérés comme des *benchmarks*. Ils ne sont pas des processus industriels, mais des systèmes simplifiés permettant de mettre en valeur ou de tester des algorithmes de pilotage avant l'application sur le processus industriel. Nous pouvons mentionner l'exemple bien connu du pendule inversé [178] afin de synthétiser un régulateur PID [179]. Nous pouvons présenter le processus chimique *Continuously Stirred Tank Reactor* (CSTR) [180]. Le système CSTR a été utilisé pour développer une commande hiérarchique EMPC avec deux niveaux de commande [181], une commande MPC robuste [182], ou une approximation de la commande MPC par un réseau de neurones [183]. Un autre exemple est le système quatre réservoirs [184]-[186]. Il a été utilisé pour comparer des régulateurs MPC [187], régulateur par mode glissant [188], commande par backstepping [189], régulateur adaptatif avec une modélisation par des neurones récurrents [190].

Dans ce travail de recherche nous avons sélectionné le système quatre réservoirs. Plusieurs

processus industriels traitent du pilotage de niveau d'eau, tel que le retraitement d'eau, le générateur de vapeur d'une centrale nucléaire, des processus chimiques et la transformation alimentaire [191]. Le système possède des propriétés intéressantes, des non-linéarités, un couplage entre sous-systèmes et des contraintes inhérentes aux hauteurs d'eau et aux actionneurs [187]. Ceci le rend d'ailleurs aussi attractif pour l'enseignement de l'automatique [186]. Le système est souvent considéré comme un système de test pour évaluer un algorithme de commande [184]. Les éléments constituant le système sont : quatre réservoirs, deux pompes et deux vannes trois voies. Par ailleurs, chaque réservoir contient un orifice ou l'eau coule, du réservoir 3 vers le 1, du réservoir 4 vers le 2, puis les réservoirs 1 et 2 dans une cuve de rétention. En outre, l'intérêt du pilotage est de maintenir un certain niveau d'eau dans les quatre réservoirs, Fig.3.1.

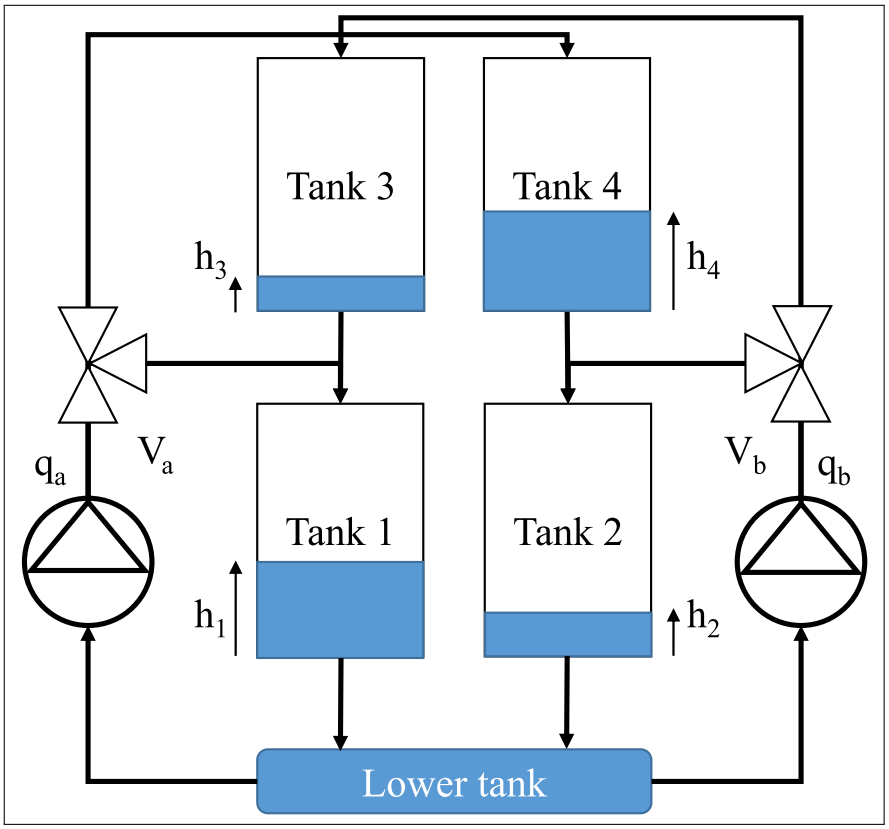

FIGURE 3.1 – Schéma du système quatre réservoirs.

Les équations physiques du système sont tirées de l'équation de Torricelli [184], [187] :

$$\dot{h}_1(t) = -\frac{a_1}{S_c}\sqrt{2g_e h_1(t)} + \frac{a_3}{S_c}\sqrt{2g_e h_3(t)} + \frac{\gamma_a}{3600 S_c} q_a(t) \tag{3.1}$$

$$\dot{h}_2(t) = -\frac{a_2}{S_c}\sqrt{2g_e h_2(t)} + \frac{a_4}{S_c}\sqrt{2g_e h_4(t)} + \frac{\gamma_b}{3600 S_c} q_b(t) \tag{3.2}$$

$$\dot{h}_3(t) = -\frac{a_3}{S_c}\sqrt{2g_e h_3(t)} + \frac{(1-\gamma_b)}{3600 S_c} q_b(t) \tag{3.3}$$

$$\dot{h}_4(t) = -\frac{a_4}{S_c}\sqrt{2g_e h_4(t)} + \frac{(1-\gamma_a)}{3600 S_c} q_a(t) \tag{3.4}$$

avec h_1, h_2, h_3, h_4 les quatre niveaux d'eau dans les réservoirs d'eau, a_1, a_2, a_3, a_4 la surface des orifices de fuite des réservoirs, S_c la section des réservoirs. q_a, q_b sont les débits des pompes et γ_a, γ_b sont les ratios d'ouverture des vannes trois voies. Les différents paramètres sont visibles à la Tab.3.1. La linéarisation du modèle non linéaire est envisageable pour obtenir le modèle linéaire [187] :

$$\dot{h}_1(t) = -\frac{a_1}{S_c\sqrt{\frac{2h_1^0}{g_e}}} h_1(t) + \frac{a_3}{S_c\sqrt{\frac{2h_3^0}{g_e}}} h_3(t) + \frac{\gamma_a}{3600 S_c} q_a(t) \tag{3.5}$$

$$\dot{h}_2(t) = -\frac{a_2}{S_c\sqrt{\frac{2h_2^0}{g_e}}} h_2(t) + \frac{a_4}{S_c\sqrt{\frac{2h_4^0}{g_e}}} h_4(t) + \frac{\gamma_b}{3600 S_c} q_b(t) \tag{3.6}$$

$$\dot{h}_3(t) = -\frac{a_3}{S_c\sqrt{\frac{2h_3^0}{g_e}}} h_3(t) + \frac{(1-\gamma_b)}{3600 S_c} q_b(t) \tag{3.7}$$

$$\dot{h}_4(t) = -\frac{a_4}{S_c\sqrt{\frac{2h_4^0}{g_e}}} h_4(t) + \frac{(1-\gamma_a)}{3600 S_c} q_a(t) \tag{3.8}$$

avec $h_1^0, h_2^0, h_3^0, h_4^0$ la hauteur d'eau considérée pour la linéarisation des réservoirs 1, 2, 3 et 4. En observant les Eq.3.5 à Eq.3.8 il est aisé d'obtenir une représentation d'état linéaire du système pour les variables d'écarts suivantes : $x_i = h_i - h_i^0$, $u_1 = q_a - q_a^0$ et $u_2 = q_b - q_b^0$. La représentation

d'état est [187] :

$$\dot{X}(t) = \begin{bmatrix} -\frac{a_1}{S_c\sqrt{\frac{2h_1^0}{ge}}} & 0 & \frac{a_3}{S_c\sqrt{\frac{2h_3^0}{ge}}} & 0 \\ 0 & -\frac{a_2}{S_c\sqrt{\frac{2h_2^0}{ge}}} & 0 & \frac{a_4}{S_c\sqrt{\frac{2h_4^0}{ge}}} \\ 0 & 0 & -\frac{a_3}{S_c\sqrt{\frac{2h_3^0}{ge}}} & 0 \\ 0 & 0 & 0 & -\frac{a_4}{S_c\sqrt{\frac{2h_4^0}{ge}}} \end{bmatrix} X(t) + \begin{bmatrix} \frac{\gamma_a}{3600S_c} & 0 \\ 0 & \frac{\gamma_b}{3600S_c} \\ 0 & \frac{(1-\gamma_b)}{3600S_c} \\ \frac{(1-\gamma_a)}{3600S_c} & 0 \end{bmatrix} U(t) \tag{3.9}$$

avec $X(t)$ le vecteur contenant les états x_1, x_2, x_3, x_4 et $U(t)$ le vecteur d'entrée de la commande u_1, u_2. Par ailleurs, les paramètres sont visibles à la Tab.3.1.

Table 3.1 – Paramètres du système quatre réservoirs [187].

Paramètres	Valeurs	Unités	Description
$h_{1_{max}}$	1.36	m	Hauteur d'eau maximale du réservoir 1
$h_{2_{max}}$	1.36	m	Hauteur d'eau maximale du réservoir 2
$h_{3_{max}}$	1.30	m	Hauteur d'eau maximale du réservoir 3
$h_{4_{max}}$	1.30	m	Hauteur d'eau maximale du réservoir 4
$h_{i_{min}}$	0.2	m	Hauteur d'eau minimale des réservoirs
$q_{a_{max}}$	3.26	m^3/h	Débit de la pompe q_a maximal
$q_{b_{max}}$	4.00	m^3/h	Débit de la pompe q_b maximal
$q_{i_{min}}$	0	m^3/h	Débit minimal des pompes
S_c	0.06	m	Section transversale des réservoirs
a_1	1.31×10^{-4}	m^2	Surface de fuite du réservoir 1
a_2	1.51×10^{-4}	m^2	Surface de fuite du réservoir 2
a_3	9.27×10^{-5}	m^2	Surface de fuite du réservoir 3
a_4	8.82×10^{-5}	m^2	Surface de fuite du réservoir 4
γ_a	0.3	-	Ratio d'ouverture de la vanne trois voies V_a
γ_b	0.4	-	Ratio d'ouverture de la vanne trois voies V_b
g_e	9.81	m/s^2	Pesanteur terrestre
h_1^0	0.65	m	Point de linéarisation du réservoir 1
h_2^0	0.66	m	Point de linéarisation du réservoir 2
h_3^0	0.65	m	Point de linéarisation du réservoir 3
h_4^0	0.66	m	Point de linéarisation du réservoir 4
q_a^0	1.63	m^3/h	Point de linéarisation du débit de la pompe q_a
q_b^0	2.00	m^3/h	Point de linéarisation du débit de la pompe q_b

3.2 Application de la méthodologie

Dans la section précédente, le système quatre réservoirs a été sélectionné pour appliquer et évaluer la méthodologie. Pour rappel, la synthèse de la méthodologie développée dans ce travail de recherche est exposée à la section 2.1. Les cinq étapes sont appliquées avec le système quatre réservoirs : modélisation fine multiphysique, génération et récoltes des données, paramétrisation du modèle boîte noire, synthèse du régulateur et évaluation du pilotage.

3.2.1 Modélisation fine multiphysique

Le système quatre réservoirs a été modélisé finement avec le langage Modelica [88] et l'outil Dymola de Dassault Systèmes. De ce fait, les équations physiques présentées n'ont pas été utilisées, Fig.3.2. Les réservoirs, tuyaux et fluides ont été réalisés avec la librairie standard Modelica, puis les pompes par la librairie Building [90]. La modélisation fine autorise la prise en compte des propriétés de l'eau et l'inertie des pompes. Les paramètres présentés dans la Tab.3.1 sont renseignés dans les sous-systèmes. Par ailleurs, pour éviter l'arrêt prématuré de la simulation, des trop-pleins sont ajoutés sur les quatre réservoirs. La modélisation Modelica possède un total de 1 049 équations et variables inconnues. Il n'est pas possible d'utiliser directement les équations pour concevoir le régulateur MPC, pour plusieurs raisons. Tout d'abord il serait nécessaire de coder ces équations dans un outil de modélisation de problème d'optimisation pour être résolues par un solveur. Il n'existe actuellement pas d'outil permettant de réaliser cette transformation automatiquement. Par ailleurs, le modèle possède des conditions if-then-else qui sont difficilement utilisables avec la MPC, sans recourir à une MPC hybride qui augmente grandement la complexité [192]. Une solution est de linéariser le modèle à un point de fonctionnement pour obtenir un couple de données d'entrées sorties du système. Puis dans un deuxième temps le modèle de représentation d'état linéaire est identifié pour obtenir les matrices. Cette méthode est utilisée dans les travaux [193] et [194]. Cette méthode n'est pas explorée dans ce travail et nous avons choisi d'identifier un modèle non linéaire par réseau de neurones.

3.2.2 Génération et récoltes des données

Afin de réaliser l'identification du système par un réseau de neurones, les états et entrées de commande ont été sélectionnés. Les états correspondent aux quatre hauteurs d'eau et les entrées aux débits des deux pompes. Des signaux ont ensuite été appliqués sur les commandes

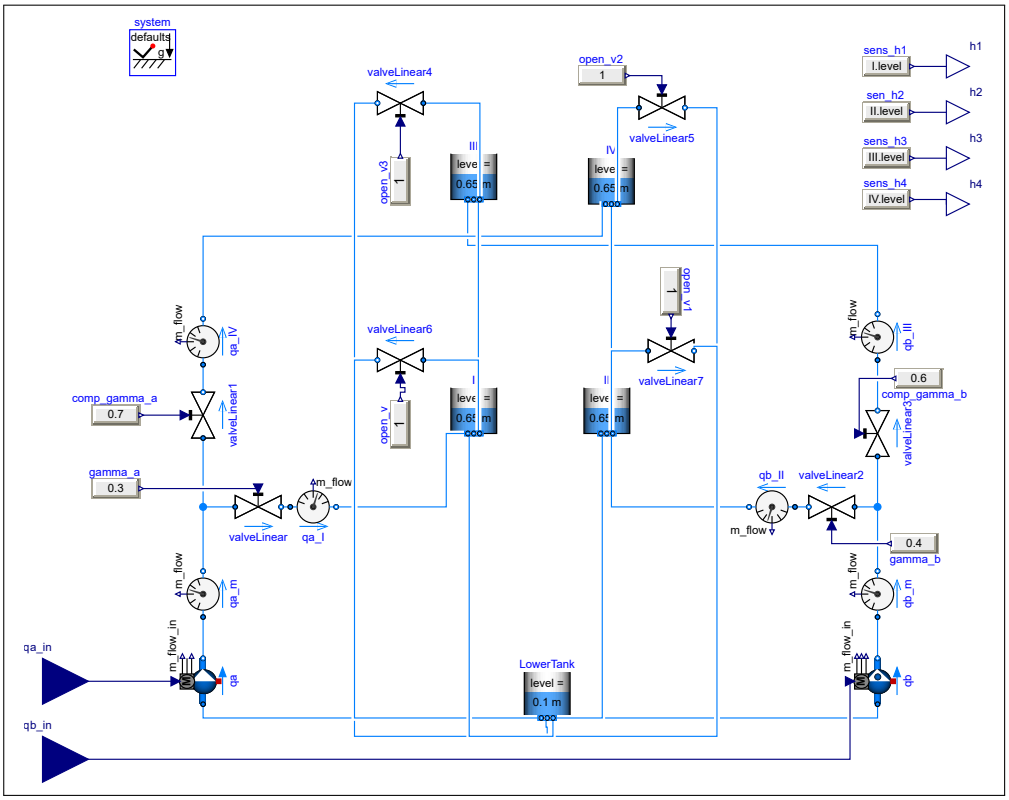

FIGURE 3.2 – Modélisation fine du système 4 réservoirs avec Dymola.

des pompes puis les hauteurs d'eau ont été mesurées. Les signaux appliqués sont des séquences binaires pseudoaléatoires, puis des échelons aléatoires entre les débits maximaux et minimaux des pompes. Les paramètres des signaux sont visibles sur les Tab.3.2. Des exemples de séries temporelles du niveau d'eau et des débits des pompes sont visibles à la Fig.3.3.

Table 3.2 – Paramètres des séquences.

Paramètres	Séquences binaires pseudo-aléatoires	Echelons aléatoires	Unités
Période de temps	500	1200	s
Temps de démarrage	500	30	$Jours$
Temps de fin	30	450	$Jours$
Amplitude	q_{min} ou q_{max}	q_{min} à q_{max}	m^3/h
Période d'échantillonnage	5	5	s

Afin de comparer et évaluer les réseaux Fnn, ResNet et Polynet, un prétraitement des données brutes issues des simulations numériques a été réalisé. Tout d'abord les données brutes de simulation sont séparées en trois groupes : données d'entrainement, données de validation et données de test. Les données d'entrainement sont utilisées pour modifier les seuils et les poids du réseau. Les données de validation sont utilisées pour l'optimisation des hyperparamètres. Puis les données de test sont utilisées pour évaluer la généralisation du réseau. Dans notre cas les données de test permettent d'évaluer la performance d'extrapolation des réseaux. Classiquement les données sont séparées aléatoirement dans l'état disponible et admissible, ce qui n'a pas été retenu ici. A la place, les données sont séparées selon les hauteurs d'eau des réservoirs 1 et 2. La séparation des données est visible à la Fig.3.4. Le processus de traitement est réalisé comme suit. Tout d'abord les données sont traitées pour conserver des séquences de 32 échantillons avec les hauteurs d'eau des quatre réservoirs [1]. Puis, les données d'entrainement et de validation sont sélectionnées lorsque les hauteurs d'eau des réservoirs 1 et 2 sont entre $0.4m$ et $0.8m$ (toute la séquence de 32 échantillons doit être dans cette zone de hauteur d'eau). Les données qui n'appartiennent pas à ce groupe constituent les données de test. (hauteur d'eau de h_1 et h_2 en dehors de $0.4m$ et $0.8m$). Ensuite, les données du premier groupe sont séparées en deux : les données d'entrainement et de validation. Les données de validation sont sélectionnées aléatoirement pour constituer 5% du volume des données. Une illustration avec un résumé du processus du traitement des données est visible à la Fig.3.5.

1. 32 est un compromis entre la taille du batch et le temps nécessaire pour l'apprentissage avec les limites calculatoires de la machine.

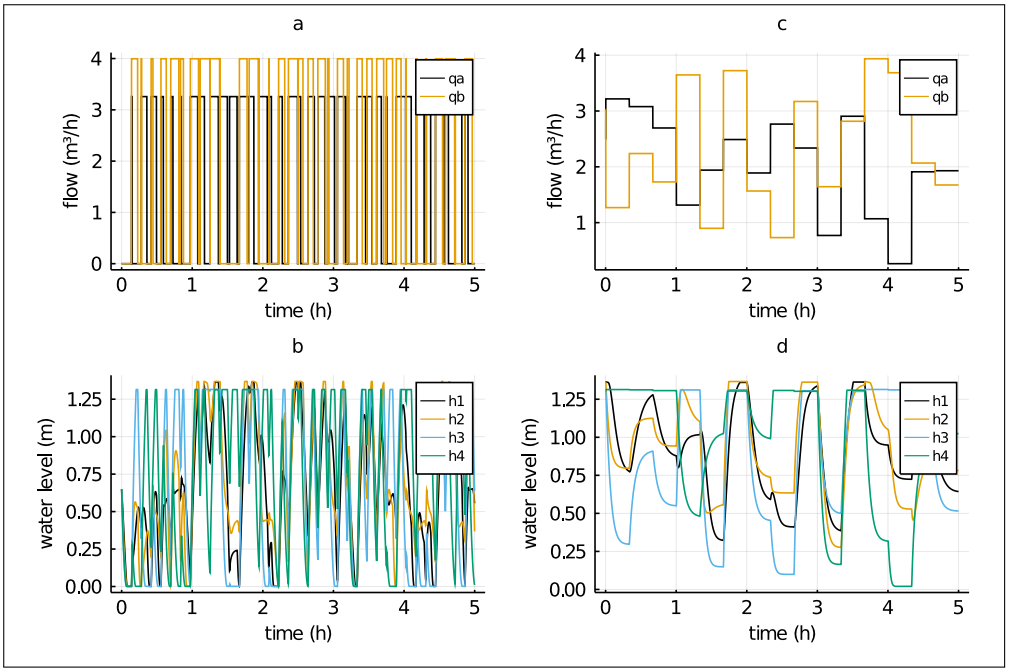

FIGURE 3.3 – Portion de données générées par les simulations Dymola. (a) Présente les signaux des entrées de commande avec des séquences binaireq pseudo-aléatoires. (b) Présente les hauteurs d'eau des réservoirs associées. (c) Présente les signaux des entrées de commande avec des échelons aléatoires. (d) Présente les hauteurs d'eau des réservoirs associées.

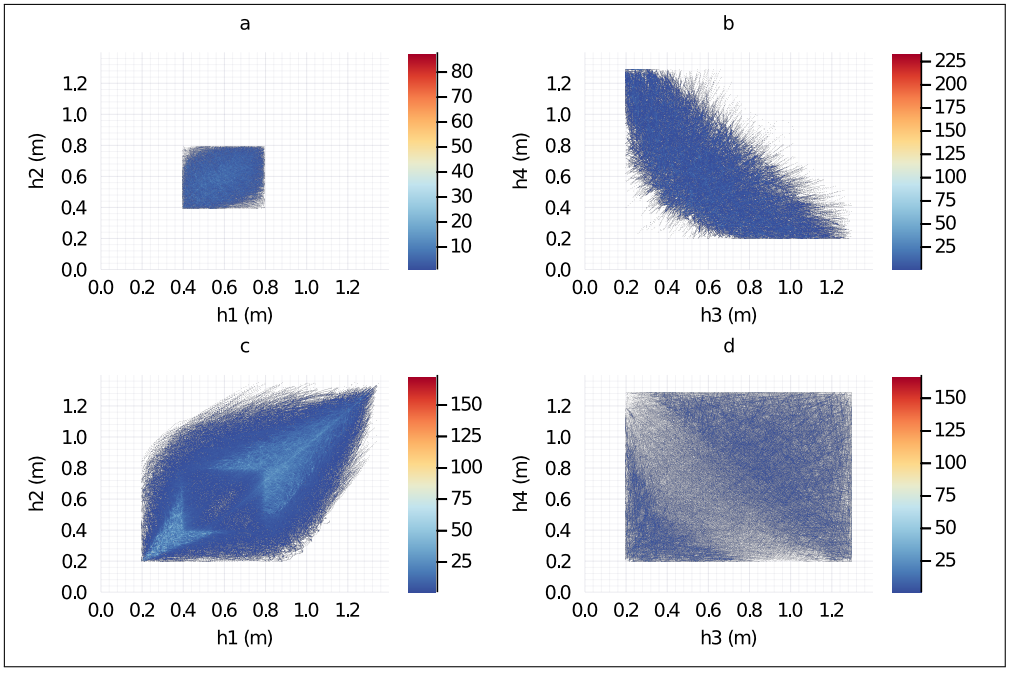

FIGURE 3.4 – Séparation des données selon la hauteur d'eau des réservoirs 1 et 2. (a) Présente les hauteurs d'eau de h_1, h_2 entre 0.4m to 0.8m. Ces données constituent les données d'entrainement et de validation. (b) Présente les hauteurs d'eau de h_3, h_4 avec les pairs de h_1, h_2 à partir des données de (a). Ces données constituent les données d'entrainement et de validation. (c) Présente les hauteurs d'eau de h_1, h_2 lorsque les hauteurs d'eau sont en dehors de 0.4m et 0.8m. Ces données constituent les données de test. (d) Présente les hauteurs d'eau de h_3, h_4 associées aux pairs de h_1, h_2. Ces données constituent les données de test.

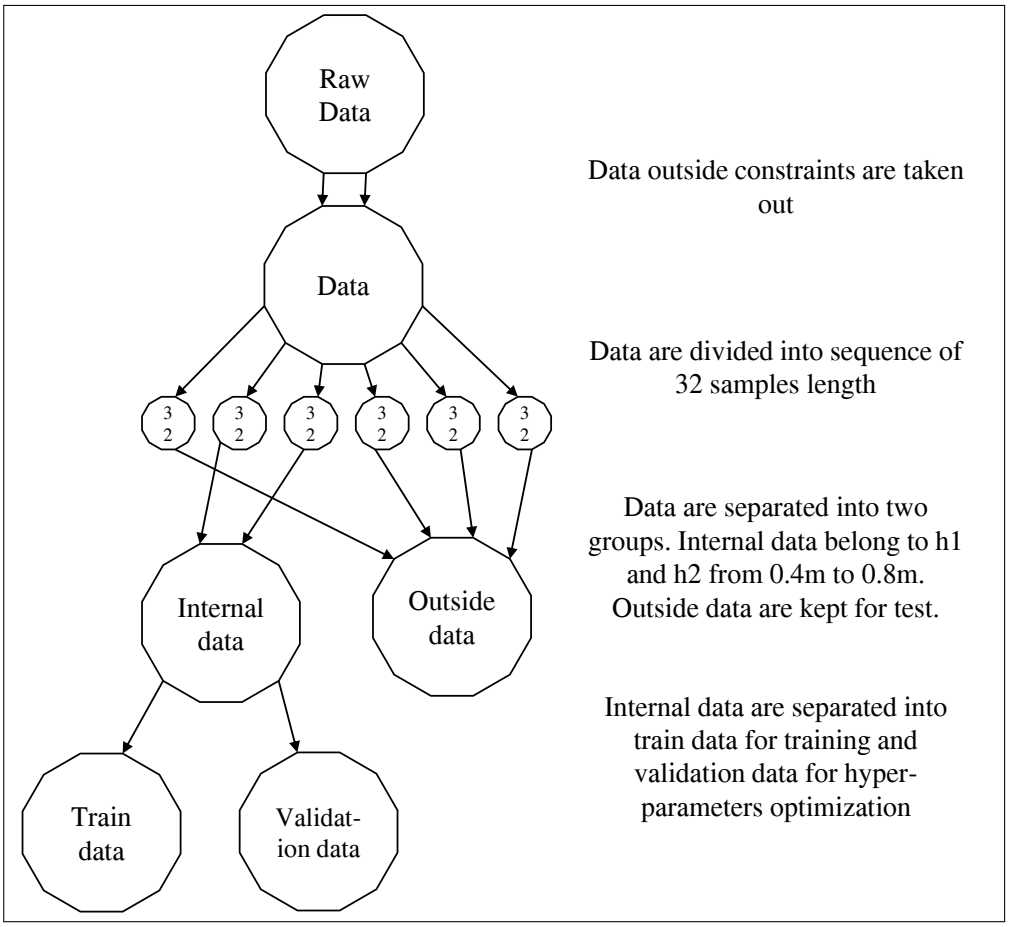

FIGURE 3.5 – Procédure de pré-traitement des données à partir des données brutes pour obtenir les données d'entrainement, de validation et de test.

Afin d'exploiter les données pour la paramétrisation du modèle, un pré-traitement est réalisé avec un couplage des données avec les données d'entrée $\mathcal{D}_e \in \{\bar{x}_k, \bar{u}_k\}$ et les données de sortie $\mathcal{D}_s \in \{\bar{x}_{k+1}\}$.

3.2.3 Paramétrisation du modèle boîte noire

Le modèle boîte noire est employé pour réaliser les prédictions des états du système au prochain instant d'échantillonnage $k+1$, tel que :

$$\hat{x}[k+1] = g_{nn}(\bar{x}[k], \bar{u}[k]) \tag{3.10}$$

avec $\hat{x}[k+1]$ le vecteur d'état prédit par le modèle boîte noire. $\bar{x}[k]$ et $\bar{u}[k]$ les vecteurs d'état et d'entrée. de commande provenant des données. g_{nn} est le modèle à paramétrer. Les trois réseaux considérés dans cette section sont le Fnn, le ResNet et le PolyNet. Chaque couche du réseau ResNet ou PolyNet est constituée d'un sous-réseau Fnn identique pour chaque couche. Par ailleurs, les couches d'entrée des réseaux sont constituées de neurones avec une fonction d'activation identité, voire chapitre 2 pour plus de détails.

Plan d'expérience

Le modèle boîte noire est calibré par l'algorithme de rétropropagation du gradient et les hyperparamètres sont optimisés avec un algorithme métaheuristique. La boîte à outils utilisée pour le modèle boîte noire est Flux [136] et l'optimisation des hyperparamètres est réalisé à l'aide de la librairie BlackBoxOptim [133] dans le langage Julia [165]. La fonction coût sélectionnée est l'erreur quadratique moyenne, voir section 2.4.4. Les paramètres autorisés influençant l'apprentissage par l'algorithme métaheuristique sont visibles à la Tab 3.3. L'optimisation et l'apprentissage est arrêté dès que 1000 réseaux sont entrainés pour chacune des architectures soit un total de 3000. Chaque itération autorise l'apprentissage de 8 réseaux en parallèle. À chaque itération l'algorithme a la possibilité de tester puis évaluer les hyperparamètres occasionnant une réduction de la fonction coût utilisée, l'erreur quadratique moyenne. Par ailleurs, l'entrainement des réseaux est réalisé par le CPU dans ce cas. En outre, après l'application de l'algorithme et entrainement des réseaux, un traitement est réalisé pour enlever les réseaux avec une fonction coût g_{loss} de l'équation (2.21) plus grande que 1. Après ce traitement, il reste 918 réseaux avec l'architecture Fnn, 964 réseaux avec l'architecture ResNet et 966 réseaux avec l'architecture PolyNet. Les résultats de la paramétrisation des modèles et de l'évaluation des trois architectures sont présentés dans la section suivante.

TABLE 3.3 – Hyperparamètres acceptables du réseau de neurones.

Paramètres	Valeurs autorisées
Fonctions d'activations	Softsign [195], Swish [196], Tanh [195], Sigmoïde [195], Fig.3.6
Neurones	4 à 20
Couches cachées	1 à 3
Couche d'intégration pour ResNet et PolyNet	1 à 9
Époques ou *Epochs*	5 à 50
Batch size	32
Optimiser	Adam [197], Radam [198], Nadam [199], Oadam [200]
Taux apprentissage	1×10^{-7} à 0.001
Momentum décroissance exponentiel	0.9 à 0.999
Momentum estimation	0.9 à 0.999

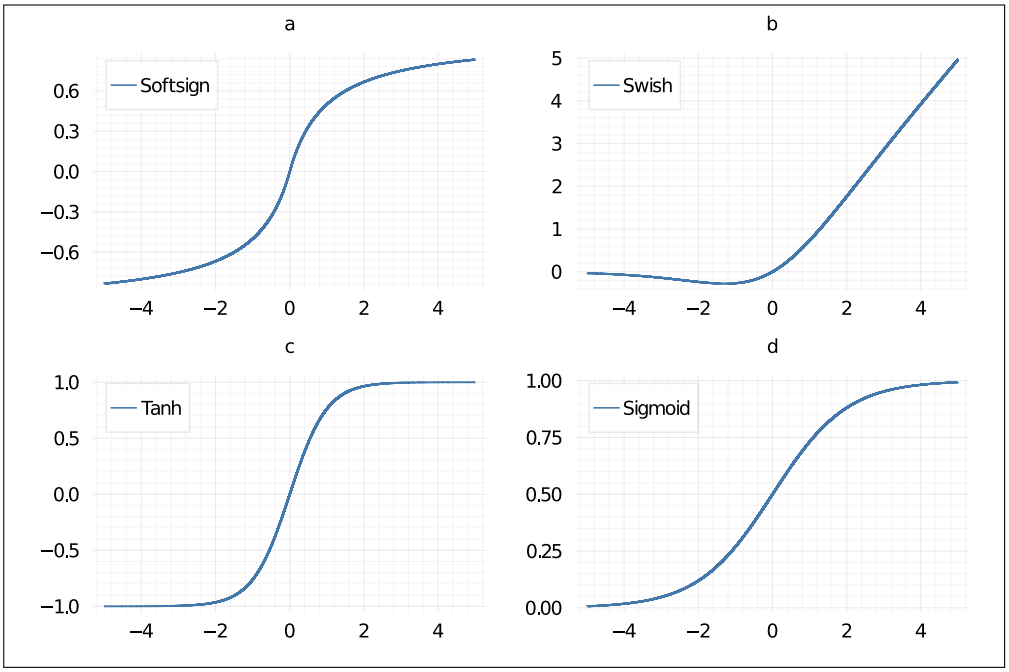

FIGURE 3.6 – Fonction d'activation, (a) Softsign : $\frac{x}{1+|x|}$, (b) Swish : $\frac{x}{1+e^{-x}}$, (c) Tanh : $tanh(x)$ et (d) Sigmoïde $\frac{1}{1+e^{-x}}$.

Résultats

La Fig.3.7 présente le résultat de l'apprentissage des réseaux avec des boîtes à moustache sur la fonction coût MSE pour les données d'entrainement, de validation et de test selon les réseaux Fnn (bleu), ResNet (rouge) et PolyNet (vert). Tout d'abord nous pouvons observer les boîtes à moustache de la fonction de coût avec les données d'entrainement, Fig.3.7.a avec les valeurs numériques présentées dans le Tab.3.4. En observant le premier quantile, la fonction coût est réduite pour les réseaux reliés à une méthode d'intégration par rapport à Fnn. En plus, la plus faible MSE est observée pour le réseau ResNet pour le premier quantile, la médiane et le dernier quantile par rapport à PolyNet et Fnn. Concernant la médiane, Fnn permet de réduire la fonction de coût par rapport à PolyNet. La même observation est visible avec les données de validation, Fig.3.7.b et Tab.3.5. Cependant, il est à noter que les performances des trois réseaux restent semblables avec les données d'entrainement et de validation. En observant les boîtes à moustache avec les données de test, Fig.3.7.b et Tab.3.6 la hiérarchie des fonctions de coût est modifiée. Dans ce cas, les deux réseaux liés aux méthodes d'intégration réalisent une réduction importante de la fonction coût pour la médiane, le premier et le quatrième quartile par rapport à Fnn.

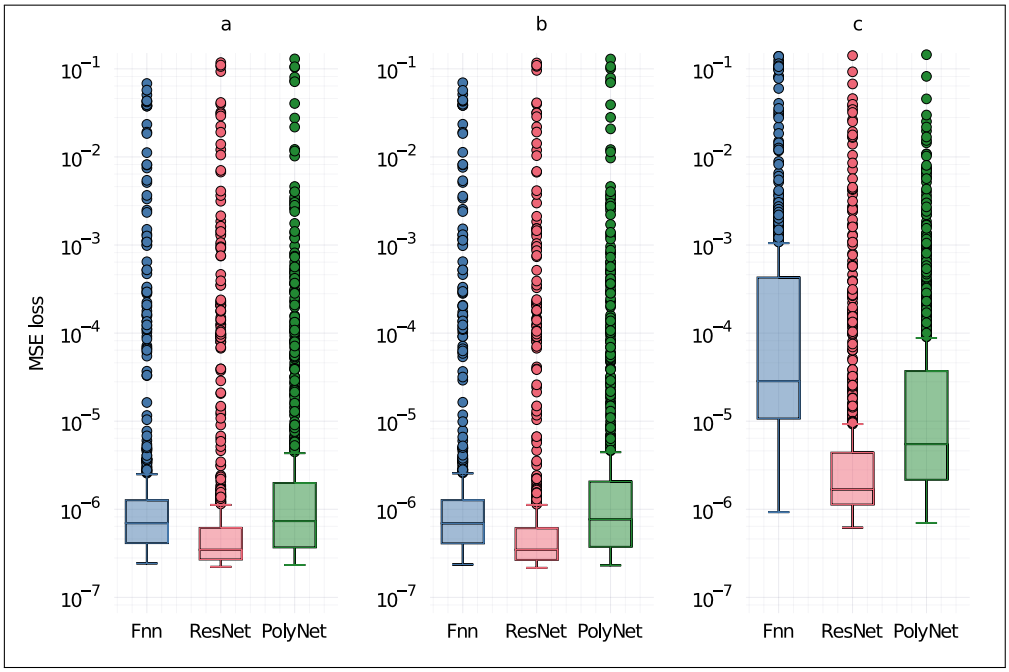

FIGURE 3.7 – boîte à moustache de la fonction de coût selon les réseaux Fnn, ResNet et PolyNet, (a) avec les données d'entrainement, (b) avec les données de validation et (c) avec les données de test.

Table 3.4 – MSE loss with train data.

Architecture	Median	First quartile	Fourth quartile
FNN	6.96×10^{-7}	4.13×10^{-7}	1.267×10^{-6}
ResNet	3.47×10^{-7}	2.69×10^{-7}	6.14×10^{-7}
PolyNet	7.34×10^{-7}	3.69×10^{-7}	1.99×10^{-6}

Table 3.5 – MSE loss with validation data.

Architecture	Median	First quartile	Fourth quartile
FNN	6.92×10^{-7}	4.10×10^{-7}	1.27×10^{-6}
ResNet	3.47×10^{-7}	2.66×10^{-7}	6.10×10^{-7}
PolyNet	7.68×10^{-7}	3.74×10^{-7}	2.01×10^{-6}

Table 3.6 – MSE loss with test data.

Architecture	Median	First quartile	Fourth quartile
FNN	2.85×10^{-5}	1.07×10^{-5}	4.28×10^{-4}
ResNet	1.68×10^{-6}	1.13×10^{-6}	4.39×10^{-6}
PolyNet	5.49×10^{-6}	2.16×10^{-6}	3.71×10^{-5}

La Fig.3.8 présente les boîtes à moustache avec la fonction de coût selon les hyperparamètres avec les données de test pour les réseaux Fnn (bleu), ResNet (rouge) et PolyNet (vert). Tout d'abord, nous pouvons observer l'effet du nombre de couches cachées sur les architectures Fnn Fig.3.8.a, ResNet Fig.3.8.f et PolyNet Fig.3.8.k. Augmenter le nombre de couches cachées augmente la fonction coût médiane pour l'architecture Fnn, avec 1.34×10^{-5} pour une couche cachée, 3.32×10^{-5} pour deux couches cachées et 3.86×10^{-5} pour trois couches cachées. Alors que l'augmentation de couche permet de réduire la fonction coût médiane pour ResNet et PolyNet 2.59×10^{-6} et 7.09×10^{-5} pour une couche, 1.83×10^{-6} et 1.03×10^{-5} pour deux couches, 1.51×10^{-6} et 4.16×10^{-6} pour trois couches.

Ensuite, nous pouvons observer la fonction de coût en fonction du nombre de neurones par couche cachée avec Fig.3.8.b pour Fnn, Fig.3.8.g, pour ResNet et Fig.3.8.l pour PolyNet. L'algorithme métaheuristique a investigué un nombre de neurones de 10 à 18 pour Fnn, de 6 à 15 pour ResNet et de 4 à 20 pour PolyNet. Il est observé pour Fnn avec la Fig.3.8.b, que la fonction de coût réalise un plateau en 10 et 17 neurones. Concernant Resnet il est observé une légère décroissance entre 6 à 14 neurones puis pour PolyNet il est observé une augmentation de 5 à 20 neurones.

Les boîtes à moustache de la fonction de coût selon les fonctions d'activation sont visibles aux Fig.3.8.c, Fig.3.8.h et Fig.3.8.m. Il est observé que la fonction swish réalise une réduction de la médiane de la fonction coût avec 5.04×10^{-4} pour sigmoïde, 9.42×10^{-4} pour softsign,

1.32×10^{-5} pour swish et 1.77×10^{-4} pour tanh. À propos de ResNet, Fig.3.8.h, les médianes de la fonction de coût avec les données de test sont de 1.34×10^{-6} pour sigmoïde, 1.73×10^{-6} pour softsign, 1.60×10^{-6} pour swish et 2.86×10^{-6} pour tanh. À propos de PolyNet, Fig.3.8.m, les médianes de la fonction de coût avec les données de test sont de 2.54×10^{-6} pour sigmoïde, 1.19×10^{-5} pour softsign, 3.67×10^{-6} pour swish et 1.80×10^{-5} pour tanh. La fonction sigmoïde permet une réduction par rapport à softsign, swish, tanh pour ResNet et PolyNet.

Puis, les boîtes à moustache de la fonction de coût selon le nombre d'epoch sont présentées aux Fig.3.8.e, Fig.3.8.j et Fig.3.8.o. L'algorithme métaheuristique a investigué un nombre d'epoch de 13 à 30 pour l'architecture Fnn, de 20 à 38 pour l'architecture ResNet et de 3 à 37 pour l'architecture PolyNet. Alors que les valeurs admissibles d'epoch pour l'algorithme métaheuristique sont de 5 à 50, celui-ci n'a pas utilisé toutes les possibilités d'epoch. Concernant l'architecture Fnn, Fig.3.8.e, augmenter les epochs de 15 à 25 permet une réduction de la fonction coût, alors que de 26 à 30 l'augmentation des epochs occasionne une augmentation de la fonction coût. Concernant le réseau ResNet, l'algorithme a sélectionné une plage d'epoch très réduite entre 30 à 36 et la fonction coût est équivalente pour ces cas, Fig.3.8.j. À propos de PolyNet, l'augmentation du nombre d'epoch occasionne une réduction de la fonction de coût médiane jusqu'à 30 epochs.

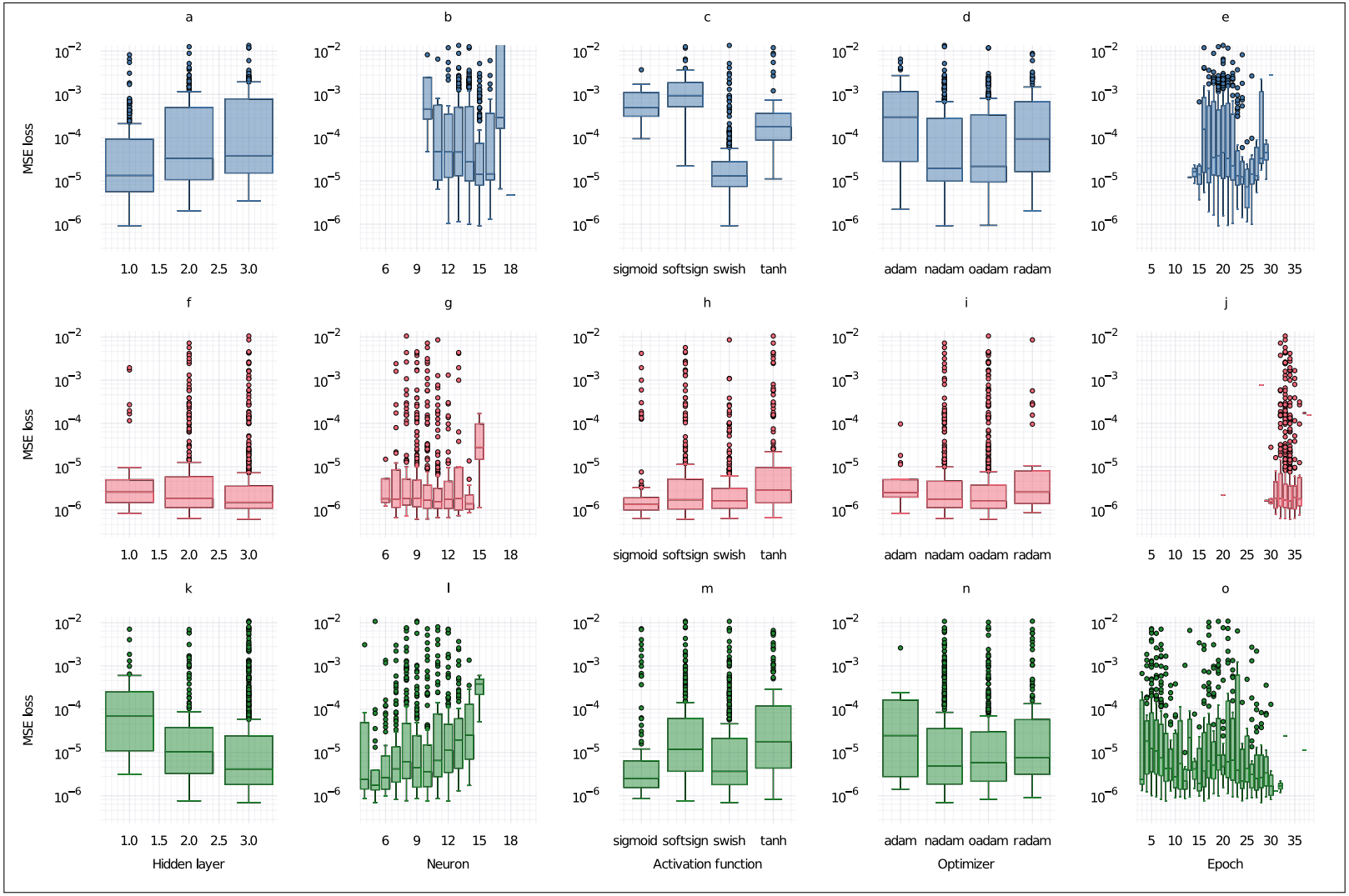

FIGURE 3.8 – Boxplot de fonctions de coût en fonction des hyperparamètres des réseaux selon, (a, f, k) le nombre de couches cachées, (b, g, i) le nombre de neurones, (c, h, m) la fonction d'activation, (d, i ,n) l'optimiseur et (e, j, o) le nombre d'époque.

Fig.3.9 expose la boîte à moustache de la fonction coût selon le nombre de couches pour les réseaux dérivés de méthode d'intégration ResNet et PolyNet avec les données de test. L'algorithme métaheuristique a investigué de 1 à 4 puis 8 couches pour ResNet, alors que pour PolyNet il a investigué de 1 à 9 couches. Il est observé sur la Fig.3.9.a que la médiane de la fonction de coût est en plateau de 1 à 4 couches avec 1.62×10^{-6} pour une couche, 1.85×10^{-6} pour deux couches, 1.54×10^{-6} pour trois couches et 1.70×10^{-6} pour quatre couches. Concernant PolyNet, visible à la Fig.3.9.b, il est observé que la fonction de coût médiane réalise un plateau de 1 à 4 couches puis augmente de 5 à 9 couches. De plus la médiane la plus faible est obtenue pour une couche avec 3.62×10^{-6}.

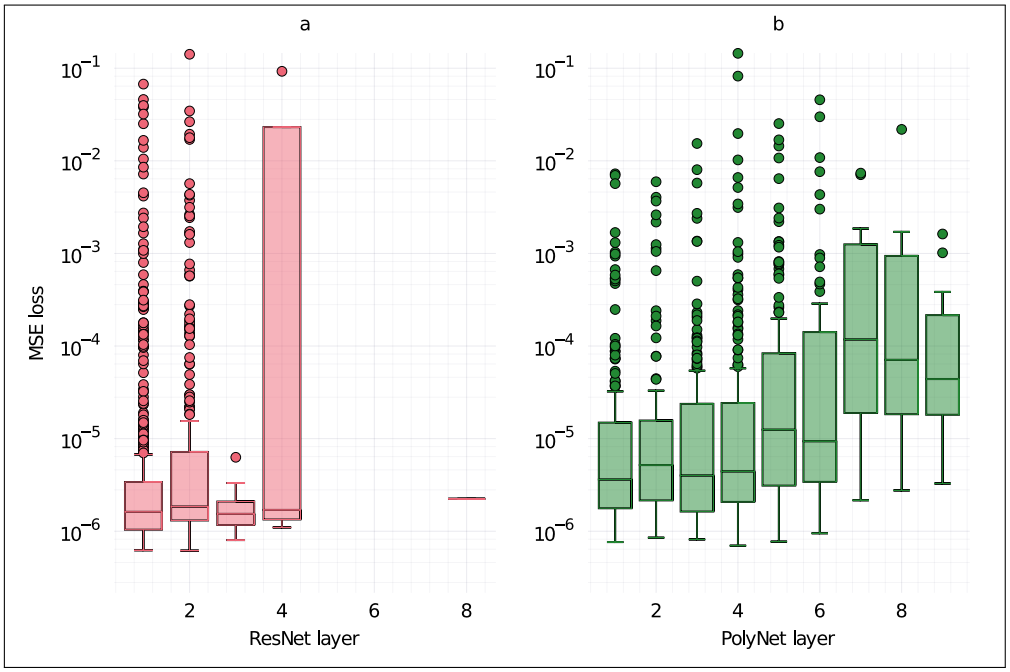

FIGURE 3.9 – Boîte à moutache avec la fonction de coût selon le nombre de couches des réseaux (a) ResNet et (b) PolyNet avec les données de test.

Fig.3.10 présente la boîte à moustache de la fonction de coût selon le temps d'entrainement avec Fnn en Fig.3.10.a, ResNet en Fig.3.10.b et PolyNet en Fig.3.10.c. Les valeurs numériques sont visibles à la Tab.3.7. Il est observé que la médiane du temps d'apprentissage est la plus faible pour Fnn avec 215.29s et la plus importante pour PolyNet avec 1118.32s, enfin ResNet est à 378.38s. Néanmoins, concernant PolyNet il est observé que le premier quantile possède un temps d'apprentissage de 265.78s alors que le premier quantile de ResNet est à 316.71s et de 161.89s pour Fnn.

Finalement, dans le but de réaliser le pilotage du système quatre réservoirs et d'évaluer les

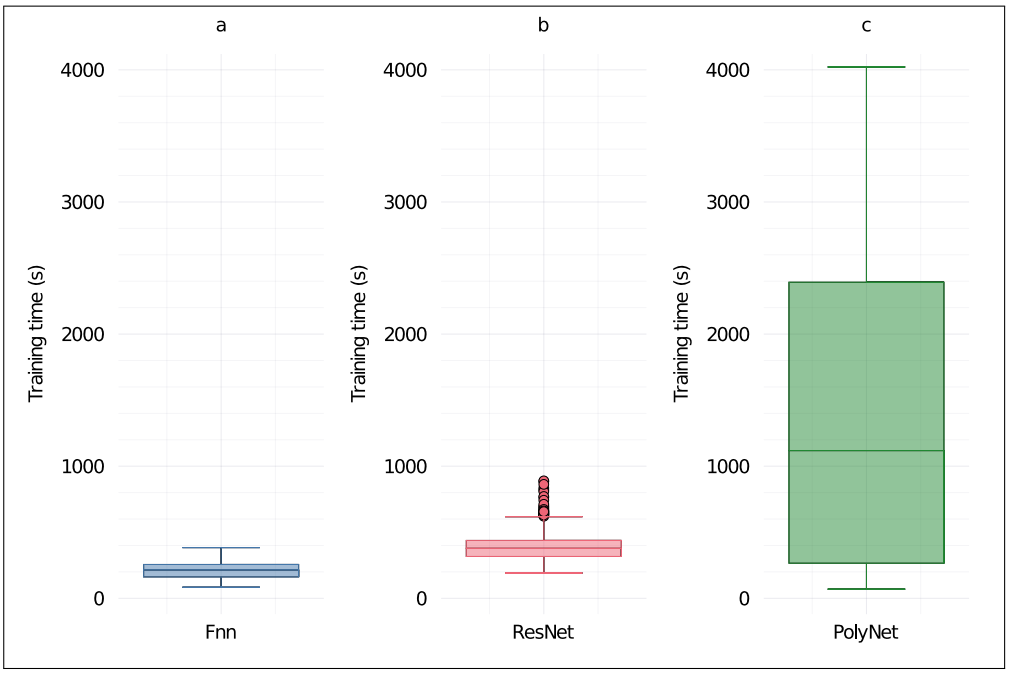

FIGURE 3.10 – boîte à moutache avec la fonction de coût selon le temps d'apprentissage des réseaux (a) Fnn, (b) ResNet et (c) PolyNet.

TABLE 3.7 – Temps d'entrainement (s).

Architecture	Mediane	Premier quantile	Quatrième quantile
Fnn	215.29	161.89	256.26
ResNet	378.38	316.71	438.17
PolyNet	1118.32	265.78	2392.48

performances de chaque architecture (Fnn, ResNet, PolyNet) avec une commande prédictive, les 25 réseaux réalisant la plus faible fonction de coût avec les données de test sont sélectionnés. Ces réseaux sont visibles aux Tab.3.8, Tab.3.9 et Tab.3.10.

TABLE 3.8 – Résultats de l'optimisation des hyperparamètres avec l'architure Fnn ayant entrainée les 25 plus faibles MSE avec les données de test.

#	Couche cachée	Neurones	Fonction activation	Epoch	Optimiseur	Train MSE	Validation MSE	Test MSE
1	1	15	swish	19	nadam	2.58×10^{-7}	2.54×10^{-7}	9.28×10^{-7}
2	1	15	swish	21	oadam	2.84×10^{-7}	2.78×10^{-7}	9.38×10^{-7}
3	1	14	swish	26	nadam	2.47×10^{-7}	2.44×10^{-7}	1.01×10^{-6}
4	1	12	swish	22	nadam	2.91×10^{-7}	2.85×10^{-7}	1.05×10^{-6}
5	1	13	swish	25	nadam	2.99×10^{-7}	2.96×10^{-7}	1.13×10^{-6}
6	1	13	swish	22	nadam	2.54×10^{-7}	2.49×10^{-7}	1.20×10^{-6}
7	1	16	swish	19	nadam	5.75×10^{-7}	5.72×10^{-7}	1.31×10^{-6}
8	1	13	swish	25	nadam	4.59×10^{-7}	4.56×10^{-7}	1.34×10^{-6}
9	1	14	swish	24	nadam	4.27×10^{-7}	4.22×10^{-7}	1.41×10^{-6}
10	1	14	swish	20	oadam	2.61×10^{-7}	2.57×10^{-7}	1.58×10^{-6}
11	1	12	swish	19	oadam	4.09×10^{-7}	4.02×10^{-7}	1.59×10^{-6}
12	1	14	swish	18	oadam	2.58×10^{-7}	2.51×10^{-7}	1.61×10^{-6}
13	1	15	swish	18	oadam	6.60×10^{-7}	6.60×10^{-7}	1.76×10^{-6}
14	1	15	swish	19	oadam	5.49×10^{-7}	5.39×10^{-7}	1.88×10^{-6}
15	1	13	swish	26	nadam	3.90×10^{-7}	3.89×10^{-7}	1.91×10^{-6}
16	1	14	swish	17	oadam	2.43×10^{-7}	2.36×10^{-7}	1.92×10^{-6}
17	1	14	swish	25	nadam	5.76×10^{-7}	5.60×10^{-7}	1.98×10^{-6}
18	1	15	swish	19	oadam	7.98×10^{-7}	7.95×10^{-7}	1.99×10^{-6}
19	1	15	swish	20	nadam	3.41×10^{-7}	3.35×10^{-7}	2.02×10^{-6}
20	1	12	swish	21	radam	4.40×10^{-7}	4.33×10^{-7}	2.04×10^{-6}
21	2	15	swish	25	oadam	3.20×10^{-7}	3.19×10^{-7}	2.05×10^{-6}
22	1	13	swish	17	nadam	3.03×10^{-7}	2.97×10^{-7}	2.06×10^{-6}
23	1	14	swish	17	oadam	2.63×10^{-7}	2.59×10^{-7}	2.09×10^{-6}
24	1	13	swish	22	nadam	4.67×10^{-7}	4.61×10^{-7}	2.11×10^{-6}
25	1	14	swish	22	oadam	2.66×10^{-7}	2.61×10^{-7}	2.16×10^{-6}

TABLE 3.9 – Résultats de l'optimisation des hyperparamètres avec l'architure ResNet ayant entrainée les 25 plus faibles MSE avec les données de test.

#	Couche cachée	Neurones	Fonction activation	Epoch	Optimiseur	Couche	Train MSE	Validation MSE	Test MSE
1	3	9	softsign	34	oadam	2	2.48×10^{-7}	2.46×10^{-7}	6.18×10^{-7}
2	3	10	softsign	34	oadam	1	2.32×10^{-7}	2.30×10^{-7}	6.23×10^{-7}
3	3	10	softsign	35	oadam	1	2.38×10^{-7}	2.33×10^{-7}	6.37×10^{-7}
4	3	10	softsign	32	oadam	1	2.34×10^{-7}	2.30×10^{-7}	6.40×10^{-7}
5	2	10	swish	33	nadam	1	2.36×10^{-7}	2.35×10^{-7}	6.45×10^{-7}
6	3	9	sigmoid	33	nadam	1	2.35×10^{-7}	2.32×10^{-7}	6.47×10^{-7}
7	3	10	sigmoid	32	nadam	1	2.38×10^{-7}	2.35×10^{-7}	6.54×10^{-7}
8	3	10	swish	34	oadam	1	2.43×10^{-7}	2.42×10^{-7}	6.55×10^{-7}
9	3	12	swish	33	oadam	1	2.42×10^{-7}	2.38×10^{-7}	6.56×10^{-7}
10	3	7	softsign	34	oadam	1	2.36×10^{-7}	2.33×10^{-7}	6.57×10^{-7}
11	3	9	tanh	34	oadam	1	2.37×10^{-7}	2.34×10^{-7}	6.69×10^{-7}
12	2	9	softsign	34	oadam	1	2.33×10^{-7}	2.29×10^{-7}	6.71×10^{-7}
13	2	10	softsign	34	oadam	1	2.35×10^{-7}	2.34×10^{-7}	6.74×10^{-7}
14	3	11	swish	32	nadam	1	2.41×10^{-7}	2.38×10^{-7}	6.74×10^{-7}
15	3	9	swish	34	oadam	1	2.37×10^{-7}	2.33×10^{-7}	6.80×10^{-7}
16	3	10	softsign	35	oadam	1	2.22×10^{-7}	2.16×10^{-7}	6.99×10^{-7}
17	2	10	softsign	34	oadam	1	2.51×10^{-7}	2.48×10^{-7}	7.08×10^{-7}
18	3	10	softsign	33	nadam	2	2.38×10^{-7}	2.32×10^{-7}	7.10×10^{-7}
19	2	10	softsign	33	nadam	1	2.40×10^{-7}	2.35×10^{-7}	7.13×10^{-7}
20	3	10	softsign	33	oadam	1	2.48×10^{-7}	2.42×10^{-7}	7.16×10^{-7}
21	2	12	softsign	32	oadam	1	2.51×10^{-7}	2.48×10^{-7}	7.18×10^{-7}
22	3	10	swish	33	oadam	1	2.36×10^{-7}	2.32×10^{-7}	7.23×10^{-7}
23	3	9	softsign	34	oadam	1	2.43×10^{-7}	2.36×10^{-7}	7.25×10^{-7}
24	3	9	softsign	34	oadam	1	2.30×10^{-7}	2.24×10^{-7}	7.27×10^{-7}
25	3	8	sigmoid	33	oadam	1	2.68×10^{-7}	2.64×10^{-7}	7.30×10^{-7}

TABLE 3.10 – Résultats de l'optimisation des hyperparamètres avec l'architure PolyNet ayant entrainée les 25 plus faibles MSE avec les données de test.

#	Couche cachée	Neurones	Fonction activation	Epoch	Optimiseur	Couche	Train MSE	Validation MSE	Test MSE
1	3	5	swish	28	nadam	4	4.18×10^{-7}	4.14×10^{-7}	6.99×10^{-7}
2	3	10	softsign	16	nadam	4	2.50×10^{-7}	2.46×10^{-7}	7.59×10^{-7}
3	2	12	swish	8	nadam	1	2.77×10^{-7}	2.74×10^{-7}	7.64×10^{-7}
4	3	8	swish	24	nadam	4	2.33×10^{-7}	2.34×10^{-7}	7.72×10^{-7}
5	3	10	softsign	16	nadam	5	2.49×10^{-7}	2.45×10^{-7}	7.73×10^{-7}
6	3	7	swish	30	nadam	3	2.38×10^{-7}	2.34×10^{-7}	8.16×10^{-7}
7	3	12	tanh	7	oadam	4	2.42×10^{-7}	2.38×10^{-7}	8.33×10^{-7}
8	3	7	swish	28	nadam	3	3.33×10^{-7}	3.32×10^{-7}	8.49×10^{-7}
9	3	9	sigmoid	12	oadam	2	2.66×10^{-7}	2.62×10^{-7}	8.53×10^{-7}
10	3	11	swish	17	oadam	3	2.56×10^{-7}	2.53×10^{-7}	8.54×10^{-7}
11	3	4	swish	25	nadam	2	2.73×10^{-7}	2.68×10^{-7}	8.69×10^{-7}
12	3	10	sigmoid	10	oadam	1	2.58×10^{-7}	2.53×10^{-7}	8.97×10^{-7}
13	3	9	sigmoid	7	radam	2	3.42×10^{-7}	3.35×10^{-7}	9.02×10^{-7}
14	3	7	swish	26	nadam	3	2.44×10^{-7}	2.40×10^{-7}	9.05×10^{-7}
15	3	5	swish	27	nadam	3	2.48×10^{-7}	2.44×10^{-7}	9.21×10^{-7}
16	3	7	swish	30	nadam	5	3.17×10^{-7}	3.13×10^{-7}	9.23×10^{-7}
17	3	8	swish	23	nadam	3	2.75×10^{-7}	2.76×10^{-7}	9.30×10^{-7}
18	3	10	sigmoid	11	oadam	1	2.92×10^{-7}	2.86×10^{-7}	9.33×10^{-7}
19	3	10	swish	17	oadam	3	2.45×10^{-7}	2.45×10^{-7}	9.35×10^{-7}
20	3	10	sigmoid	13	oadam	3	2.82×10^{-7}	2.79×10^{-7}	9.40×10^{-7}
21	2	11	softsign	6	oadam	1	2.50×10^{-7}	2.47×10^{-7}	9.45×10^{-7}
22	3	5	swish	29	nadam	3	2.42×10^{-7}	2.39×10^{-7}	9.48×10^{-7}
23	3	8	swish	30	nadam	3	3.01×10^{-7}	2.97×10^{-7}	9.50×10^{-7}
24	3	8	swish	16	nadam	6	2.53×10^{-7}	2.48×10^{-7}	9.51×10^{-7}
25	3	8	swish	22	nadam	6	2.64×10^{-7}	2.61×10^{-7}	9.60×10^{-7}

3.2.4 Synthèse du régulateur

La commande prédictive utilisée pour piloter le système quatre réservoirs est une MPC contractive. Elle possède une fonction coût quadratique et une contraction des états finaux pour assurer la stabilité et la faisabilité [152] :

$$\min_{\tilde{u},\tilde{x}} \sum_{k=0}^{N-1} \tilde{x}_k^T Q \tilde{x}_k + \tilde{u}_k^T R \tilde{u}_k \tag{3.11}$$

$$\text{s.c. } \hat{x}_{k+1} = g_{nn}^*(\hat{x}_k, u_k) \tag{3.12}$$

$$\hat{x}_k \in \mathcal{X} \tag{3.13}$$

$$u_k \in \mathcal{U} \tag{3.14}$$

$$\hat{x}_0 = \bar{x}(t) \tag{3.15}$$

$$\tilde{x}_N P \tilde{x}_N \leq \alpha \tilde{x}_0 P \tilde{x}_0 \tag{3.16}$$

$$\tilde{x}_k = \hat{x}_k - x_k^r \tag{3.17}$$

$$\tilde{u}_k = u_k - u_k^r \tag{3.18}$$

avec l' Eq.3.11 la fonction coût quadratique à minimiser. $\tilde{u}$ est la séquence d'entrée aux écarts : $\tilde{u} := \tilde{u}_0, \ldots, \tilde{u}_{N-1}$ et $\tilde{x}$ est la séquence d'état aux écarts : $\tilde{x} := \tilde{x}_0, \ldots, \tilde{x}_N$ avec N l'horizon. L'Eq.3.12 est l'équation d'évolution des états, elle est tirée du prédicteur à base de neurones artificiels. Les contraintes sur les états et les entrées de commande sont visibles aux Eq.3.13 et Eq.3.14. Eq.3.15 concerne l'initialisation des états avec la mesure des états du système avant l'exécution du problème d'optimisation pour une fenêtre temporelle donnée. Eq.3.16 présente les contraintes finales avec les contractions sur les états. Finalement, l' Eq.3.17 et l'Eq.3.18 sont les écarts des états et des entrées de commande par rapport aux consignes.

Par ailleurs, afin d'évaluer l'influence des paramètres Q, R, N et des réseaux de neurones artificiels, sur le pilotage du système, un algorithme est utilisé dans le but de minimiser l'erreur de consigne 3.11. L'index utilisé est l'intégration de cette erreur durant la simulation, avec *Integrated absolute error (IAE)* : $\int_{t_1}^{t_2} |e(t)|\, dt$ avec t_1 de 2700s et t_2 de 12000s. Fenêtre temporelle choisie ainsi, car cela permet de laisser du temps de démarrage pour l'initialisation du système dynamique (0s et 2700s) puis de laisser 3000s pour chaque scénario de consigne. Les différentes valeurs admissibles pour l'optimisation sont visibles à la Tab.3.11. L'optimisation de l'IAE est réalisée selon les contraintes des paramètres de la MPC, puis les simulations sont arrêtées lorsque plus de 150 évaluations ont été réalisées. Ces 150 évaluations sont réalisées pour les trois architectures Fnn, ResNet, Polynet puis les résultats sont présentés et analysés.

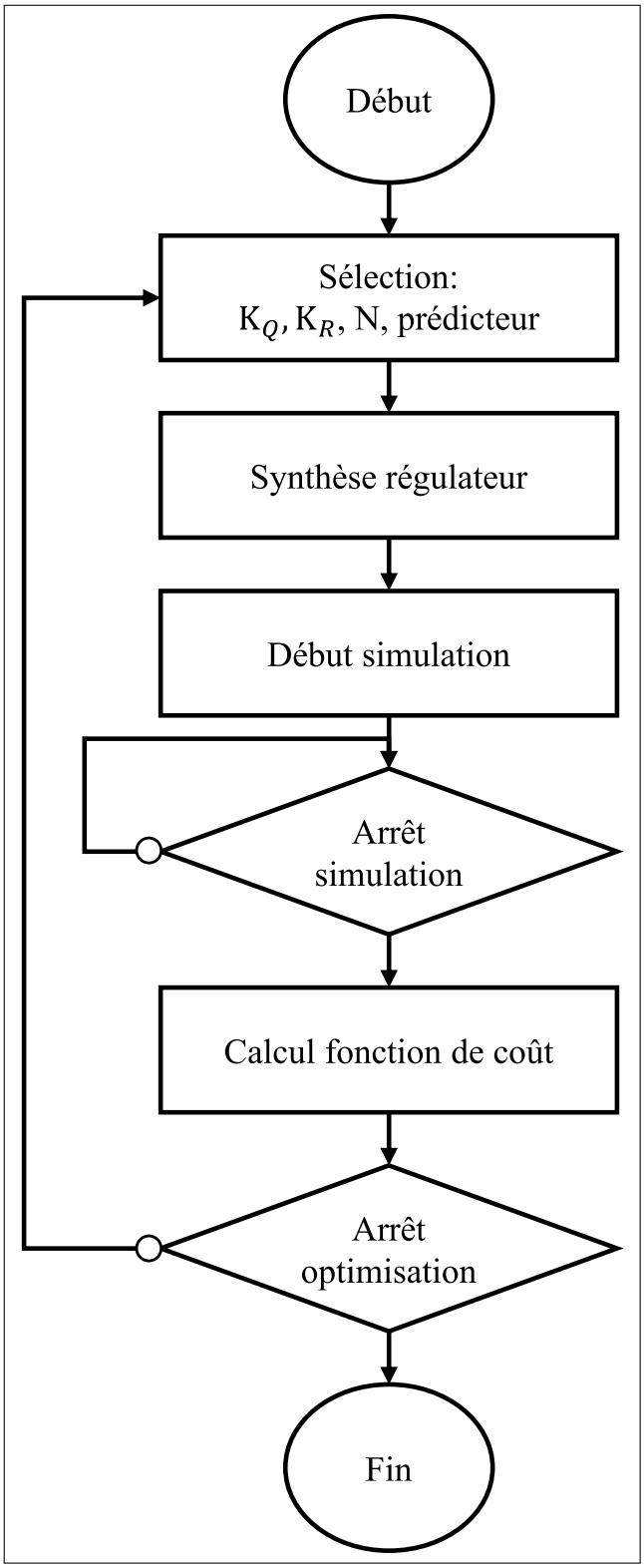

FIGURE 3.11 – Algorithme d'optimisation des paramètres de la MPC.

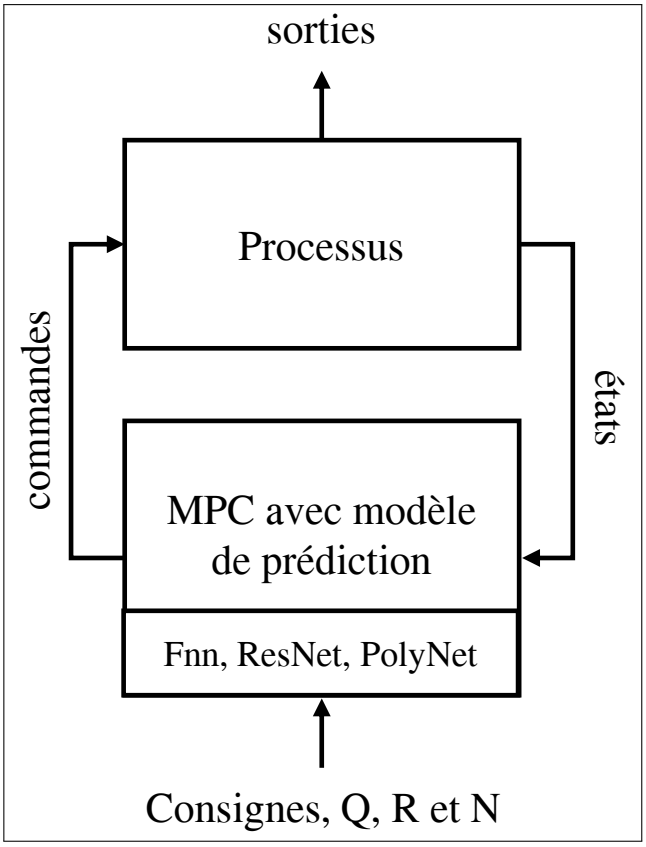

FIGURE 3.12 – Schématisation des régulateurs à base de MPC synthétisées pour piloter le système quatre réservoirs.

TABLE 3.11 – Paramètres admissibles pour l'algorithme 3.11

Paramètres	Valeurs
Q	$\kappa_Q \times$ diag(1 1 1 1)
κ_Q	1 to 10
R	$\kappa_R \times$ diag(1 1)
κ_R	0.01 to 1
N	2 to 20
α	0.90
Réseaux	Les 25 meilleurs

3.2.5 Evaluation du pilotage

Plan d'expérience

La modélisation fine du système quatre réservoirs est exportée en *Functional Mock-up Unit* (FMU) en utilisant le protocole *Functional Mock-up Interface* (FMI) [201]. La FMU est simulée dans un environnement de simulation pour réaliser les connexions logicielles avec la commande MPC développée avec JuMP dans le langage de programmation julia. Le solveur de résolution du problème d'optimisation est Ipopt [171]. L'environnement de simulation sélectionné est Simulink de Matlab pour la simplicité de mise en oeuvre. La commande MPC est calculée toutes les 5 secondes. Ainsi, la FMU est simulée, puis toutes les 5 secondes arrêtée, la commande est calculée puis appliquée sur les entrées de commande (pompes hydrauliques), enfin la simulation est relancée pour 5 secondes. Durant toute la simulation, les états et les entrées de commande sont sauvegardés pour affichage. Un schéma de l'environnement de simulation avec la FMU et la commande est visible à la Fig.3.13. Par ailleurs, les points de consignes utilisés pour le pilotage des hauteurs d'eau sont visibles à la Tab.3.12.

Résultats

La Fig.3.14 présente les résultats des simulations avec les boîtes à moustache de l'IAE selon les états et leur moyenne avec la MPC basée sur le modèle de prédictiction Fnn. Il est observé en excluant les intrus que les boîtes sont aplaties pour la moyenne ainsi que les quatre états. Nous pouvons indiquer qu'il existe un minimum de l'IAE et que l'algorithme métaheuristique a permis de l'obtenir. Nous ne pouvons pas indiquer si celui-ci est un minimum local ou un minimum global de par la nature de cet algorithme. Par ailleurs, il est observé que l'IEA sur l'état x_3 engendre un IAE plus faible par rapport aux états, x_1, x_2 et x_4.

Fig.3.15 présente pour chaque simulation réalisée l'IAE selon les paramètres κ_Q, κ_R et

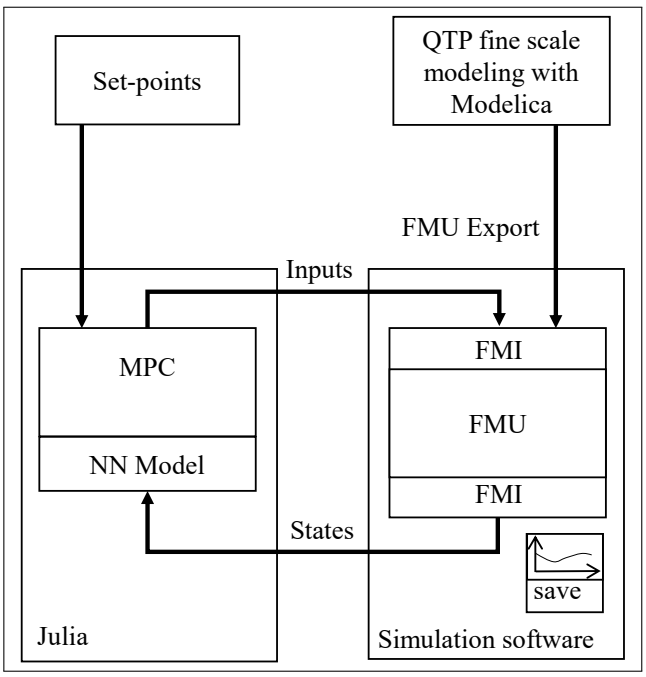

FIGURE 3.13 – Environnement numérique de simulation contenant la FMU et le régulateur MPC à base de neurones artificiels.

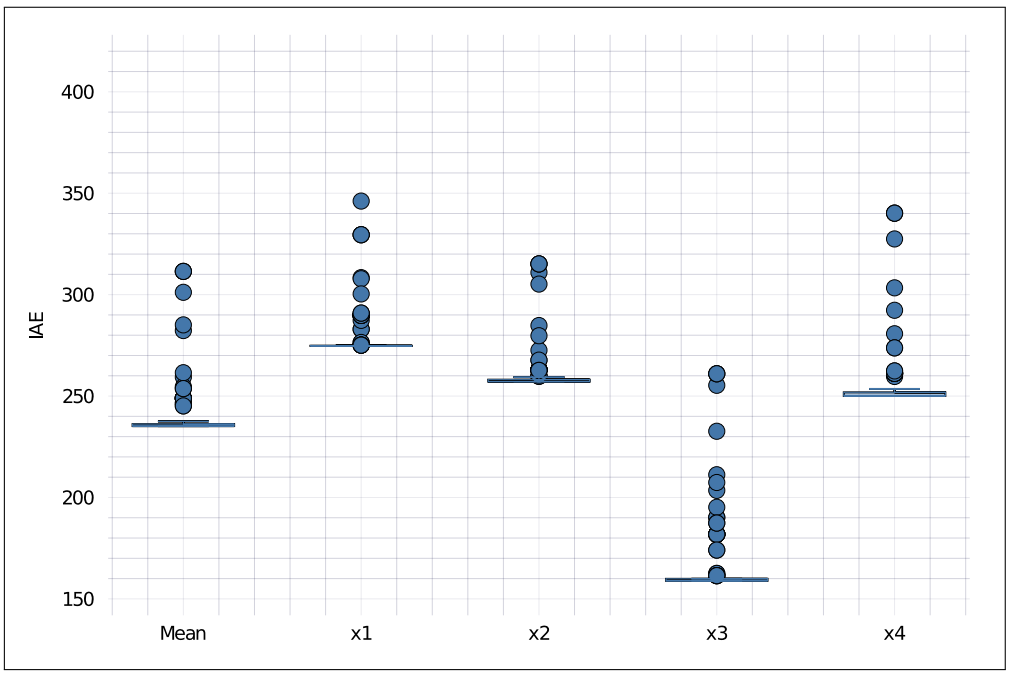

FIGURE 3.14 – boîte à moustache de l'IAE pour les 150 pilotages du système quatre réservoirs avec la MPC basée sur le modèle de prédiction Fnn.

TABLE 3.12 – Points de consigne en régime établi pour le pilotage du système quatre réservoirs avec la commande MPC.

References	Consignes des états	Consignes des entrées	Temps
Reference 1	$x_1 = 0.650$	$u_1 = 1.637$	0 à 3000
	$x_2 = 0.650$	$u_2 = 1.988$	
	$x_3 = 0.652$		
	$x_4 = 0.664$		
Reference 2	$x_1 = 0.300$	$u_1 = 1.112$	3000 à 6000
	$x_2 = 0.300$	$u_2 = 1.351$	
	$x_3 = 0.301$		
	$x_4 = 0.306$		
Reference 3	$x_1 = 0.500$	$u_1 = 2.201$	6000 à 9000
	$x_2 = 0.750$	$u_2 = 1.361$	
	$x_3 = 0.305$		
	$x_4 = 1.200$		
Reference 4	$x_1 = 0.900$	$u_1 = 1.528$	9000 à 12000
	$x_2 = 0.750$	$u_2 = 2.539$	
	$x_3 = 1.062$		
	$x_4 = 0.579$		

l'horizon N, Fig.3.15.a, Fig.3.15.b et Fig.3.15.c respectivement. En étudiant la Fig.3.15.a, nous pouvons extraire que l'algorithme a évalué plusieurs simulations avec le paramètre κ_Q entre les bornes admissibles. Néanmoins, le paramètre κ_Q qui occasionne la plus faible IAE est situé à 5.58. Ensuite, nous pouvons apprécier l'IAE en fonction du paramètre κ_R avec la Fig.3.15.b. Ainsi, entre 0.01 et 0.2, il est observé que plusieurs paramètres ont concouru à trouver le minimum de l'IAE moyen. Néanmoins, utiliser un κ_R entre 0.01 et 0.2 pour ce cas d'étude n'est pas une garantie, car des simulations n'ont pas atteint le minimum observable. De ce fait, les paramètres κ_Q et N ont aussi une responsabilité dans la réduction de l'IAE. Finalement, avec Fig.3.15.c nous pouvons observer que l'augmentation de l'horizon jusqu'à des valeurs proches du maximum admissible permet dans certain cas une réduction de l'IAE par rapport à un horizon faible. En outre, il est constaté que l'algorithme n'a pas testé beaucoup de paramètres N, celui-ci a principalement sélectionné un horizon de 19.

La Fig.3.16 expose les trajectoires de pilotage des hauteurs d'eau avec l'architecture Fnn. Fig.3.16.a expose les trajectoires de h_2 en fonction de h_1 pour les 150 simulations. Fig.3.16.b présente les trajectoires avec h_4 en fonction de h_3 pour les 150 simulations. Fig.3.16.c présente la trajectoire de h_3 en fonction de h_1 réalisant la plus faible IAE moyen et Fig.3.16.d pour la trajectoire de h_4 en fonction de h_3. Ces figures permettent de mettre en lumière des dépassements

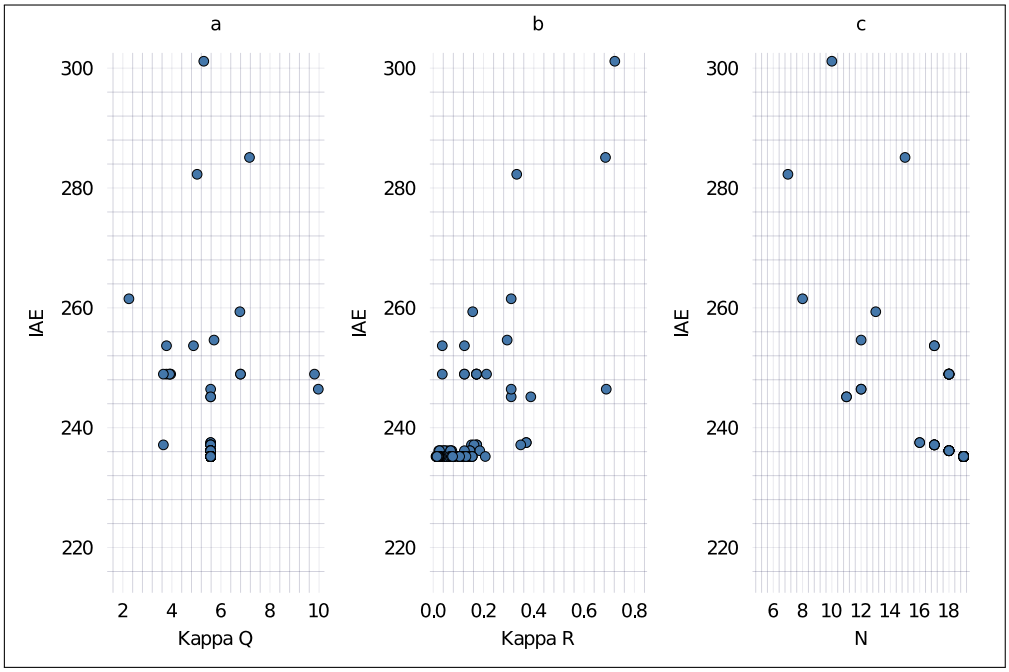

FIGURE 3.15 – IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture Fnn.

éventuels en régime établi et les trajectoires hétérogènes entre les simulations. Ainsi, des dépassements sont observés aux points de consignes sur les Fig.3.16.a et Fig.3.16.b. Nous pouvons indiquer que les paramètres de la MPC ont une influence sur la trajectoire des états et sur les dépassements en régime stabilisé. En observant les Fig.3.16.c et Fig.3.16.d, il n'est pas observé de dépassement aux points de consignes. Finalement, nous pouvons examiner les hauteurs d'eau utilisées pour l'entrainement du prédicteur Fnn, cette zone est matérialisée avec un rectangle gris sur les Fig.3.16.a et Fig.3.16.c. L'influence des données d'entrainement sur une zone réduite n'est pas visible avec les Fig.3.16.a et Fig.3.16.c. Fig.3.16.a présente des trajectoires avec des dépassements dedans qu'à l'extérieur du rectangle gris.

Fig.3.17 présente les résultats de la simulation avec la hauteur d'eau du réservoir un (Fig.3.17.a) et deux (Fig.3.17.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainés la plus faible IAE et l'architecture Fnn. Tout d'abord il est observé que les contraintes de hauteurs d'eau sont respectées pendant la simulation. Par ailleurs, quand les points de consignes changent, des dépassements des hauteurs d'eau sont visibles : Fig.3.17.a à 6000s avec h_{r1} et Fig.3.17.b à 9000s. De plus, il est observé un comportement d'un système à non minimum de phase avec la Fig.3.17.a à 9000s.

Fig.3.18 présente les résultats de la simulation avec la hauteur d'eau du réservoir un

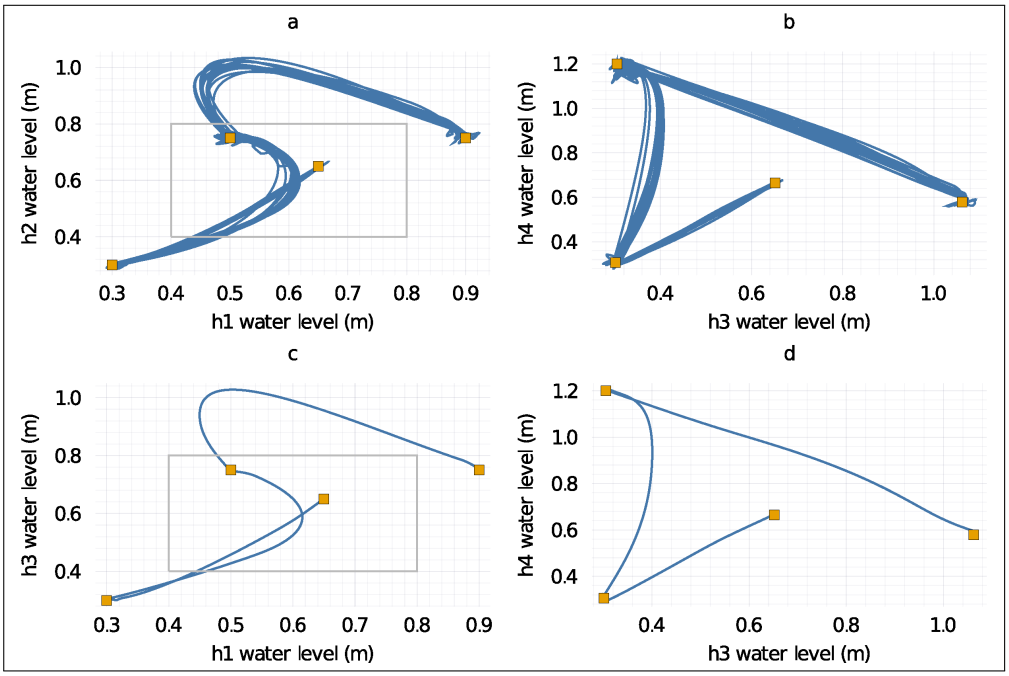

FIGURE 3.16 – Trajectoires de pilotage du système quatres réservoirs avec l'architecture Fnn. Les carrés jaune indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4.

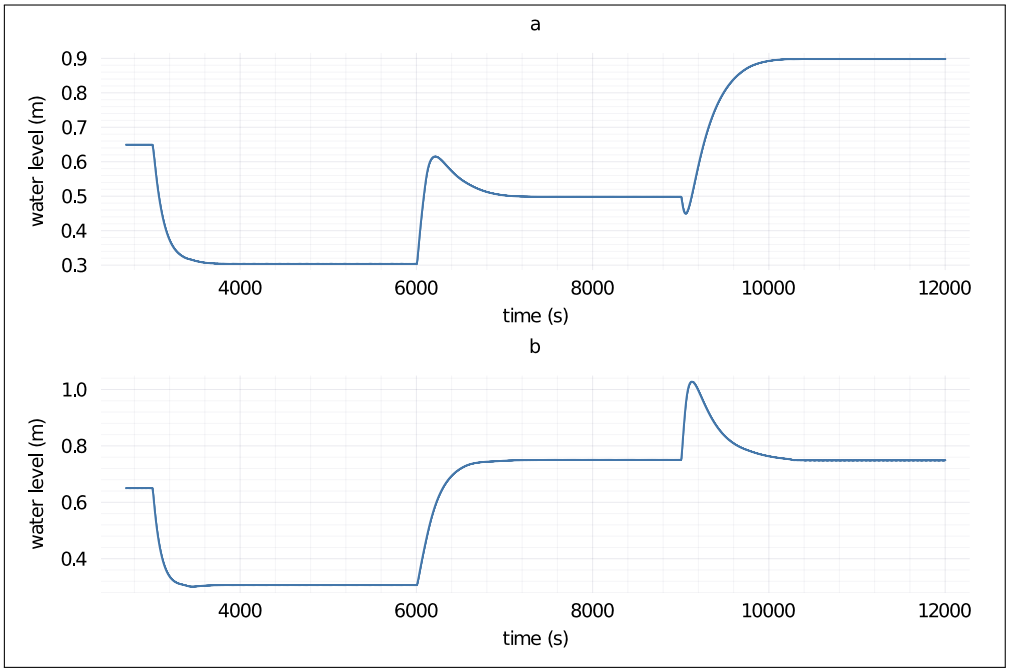

FIGURE 3.17 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture Fnn : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2.

(Fig.3.18.a) et deux (Fig.3.18.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainés la plus faible IAE et l'architecture Fnn. En premier lieu, il est observé que les hauteurs d'eau des deux réservoirs restent dans les plages admissibles des hauteurs d'eau. Par ailleurs, il y a un dépassement visible à 6000s sur la hauteur d'eau du réservoir h_{r1}, Fig.3.18.a.

Fig.3.19 présente les résultats de la simulation avec le débit d'eau de la pompe q_a (Fig.3.19.a) et de la pompe q_b (Fig.3.19.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainé la plus faible IAE et l'architecture Fnn. Nous pouvons observer que le débit des deux pompes reste dans les contraintes admissibles. De plus, les contraintes maximales sont atteintes quand il y a un changement de point de consigne (Fig.3.19.a à 3000s, 6000s, 9000s et Fig.3.19.b à 3000s, 6000s, 9000s).

La Fig.3.20 présente les résultats des simulations avec les boîtes à moustache de l'IAE selon les états et leur moyenne avec la MPC basée sur le modèle de prédictiction ResNet. Il est observé en excluant les intrus, que les boîtes sont plus larges que l'évaluation avec l'architecture Fnn. De ce fait, les différentes simulations réalisées par l'algorithme engendrent un IAE variant plus fortement que les simulations réalisées par le modèle Fnn. Par ailleurs, il est observé que l'IAE

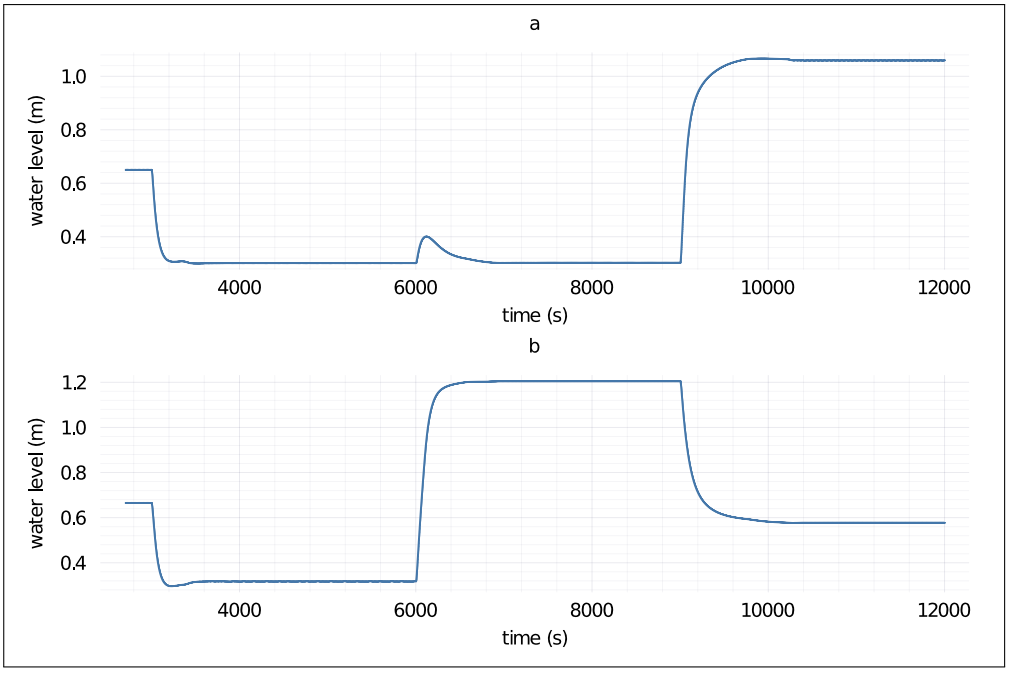

FIGURE 3.18 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture Fnn : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4.

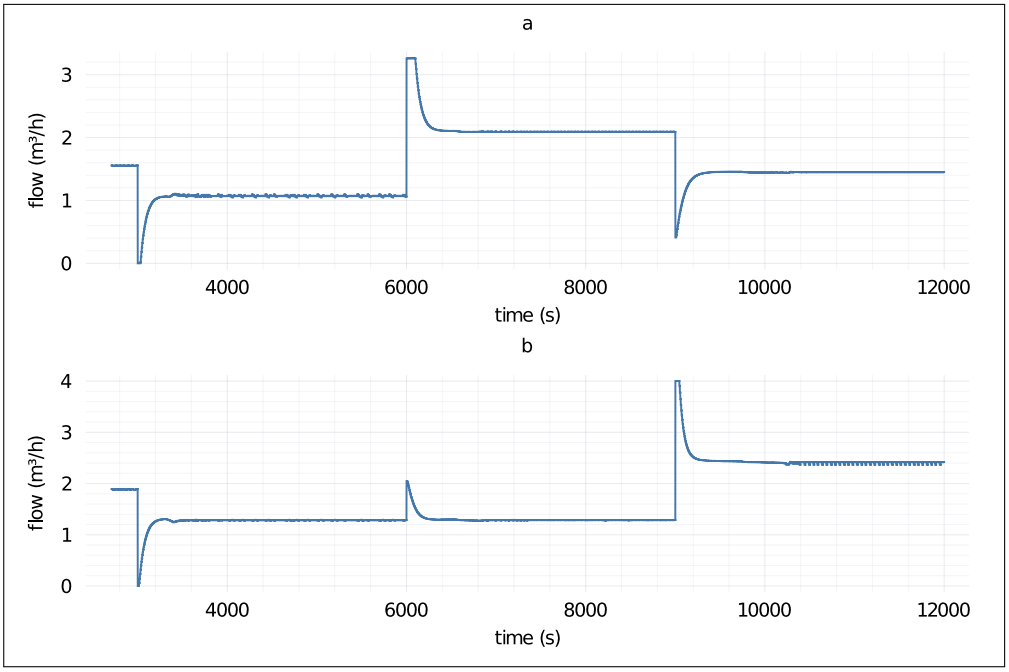

FIGURE 3.19 – Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture Fnn : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b.

sur l'état x_3 engendre un IAE plus faible par rapport aux états, x_1, x_2 et x_4.

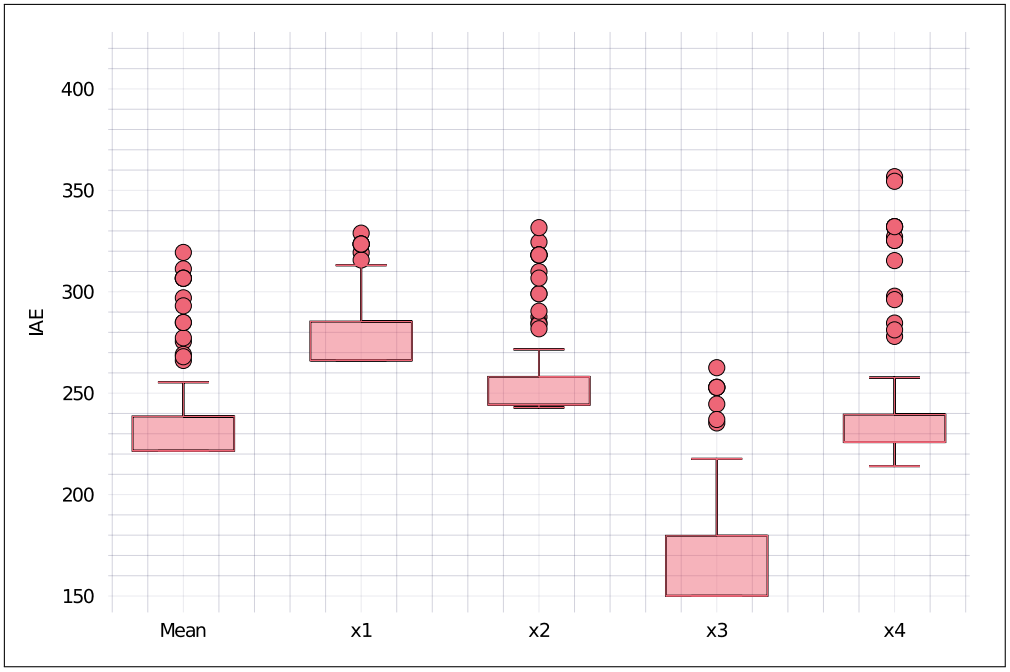

FIGURE 3.20 – boîte à moustache des 150 pilotages du système quatre réservoirs avec l'architecture ResNet et les paramètres de la MPC. L'IAE est présentée en fontion des états du système.

La Fig.3.21 présente pour chaque simulation réalisée l'IAE selon les paramètres κ_Q, κ_R et l'horizon N, Fig.3.21.a, Fig.3.21.b et Fig.3.21.c respectivement. En étudiant la Fig.3.21.a, nous pouvons extraire que l'algorithme a évalué plusieurs simulations avec le paramètre κ_Q entre les bornes admissibles. Les résultats des simulations montrent que plusieurs paramètres κ_Q entrainent l'obtention de l'IAE minimum, en effet plusieurs points atteignent un plateau bas. Ensuite, nous pouvons apprécier l'IAE en fonction du paramètre κ_R avec la Fig.3.21.b. Ainsi, entre 0.01 et 0.2, il est observé que plusieurs paramètres ont concouru à trouver le minimum de l'IAE moyen. Néanmoins, utiliser un κ_R entre 0.01 et 0.2 pour ce cas d'étude n'est pas une garantie, car des simulations n'ont pas atteint le minimum observable, de ce fait les paramètres κ_Q et N ont aussi une responsabilité dans la réduction de l'IAE. Finalement, avec Fig.3.21.c nous pouvons observer que l'augmentation de l'horizon jusqu'à des valeurs proches du maximum admissible permet dans certain cas une réduction de l'IAE par rapport à un horizon faible.

Fig.3.22 affiche les trajectoires de pilotage des hauteurs d'eau avec l'architecture ResNet. Fig.3.22.a expose les trajectoires de h_2 en fonction de h_1 pour les 150 simulations. Fig.3.22.b présente les trajectoires avec h_4 en fonction de h_3 pour les 150 simulations. Fig.3.22.c présente la trajectoire de h_3 en fonction de h_1 réalisant la plus faible IAE moyen et la Fig.3.22.d pour la

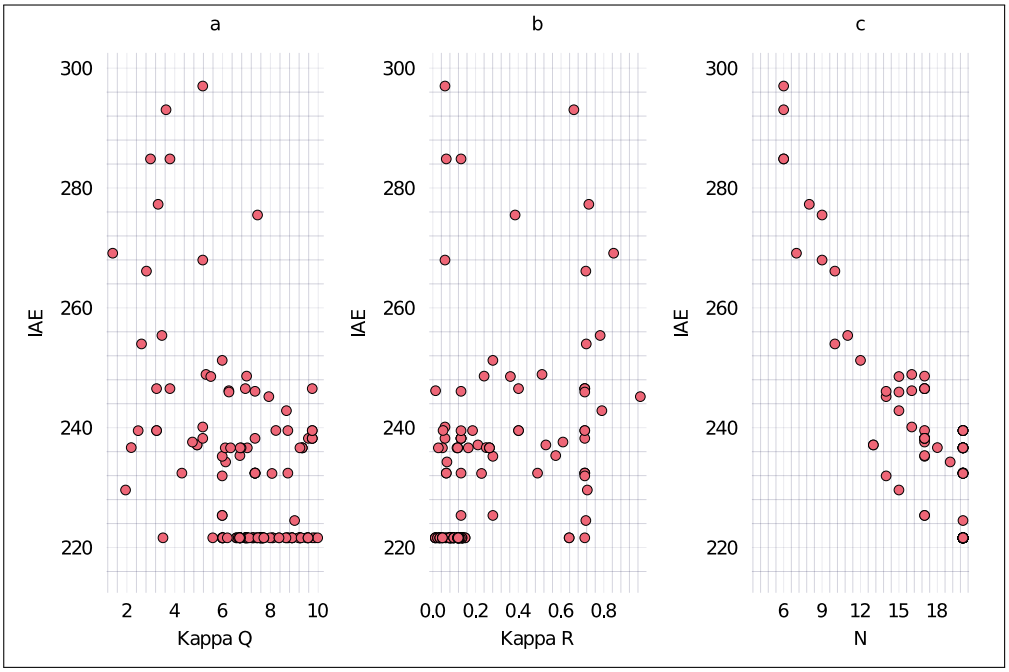

FIGURE 3.21 – IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture ResNet.

trajectoire de h_4 en fonction de h_3. Ces figures permettent de mettre en lumière des dépassements éventuels en régime établi et les trajectoires hétérogènes entre les simulations. Ainsi, des dépassements sont observés aux points de consignes sur la Fig.3.22.b pour le point de référence 4, Tab.3.12. Cet élément est une amélioration, car les dépassements sont observés à chaque référence avec l'architecture du prédicteur Fnn. Concernant, la trajectoire entrainant la plus faible IAE il n'est pas observé de dépassement aux points de références, Fig.3.22.c et Fig.3.22.d. Puis, nous pouvons examiner les hauteurs d'eau utilisées pour l'entrainement du prédicteur ResNet, cette zone est matérialisée avec un rectangle gris sur les Fig.3.22.a et Fig.3.22.c. L'influence des données d'entrainement sur une zone réduite n'est pas visible avec les Fig.3.22.a et Fig.3.22.c car il n'est pas présenté de trajectoires avec des dépassements aux points de consignes.

La Fig.3.23 présente les résultats de la simulation avec la hauteur d'eau du réservoir un (Fig.3.23.a) et deux (Fig.3.23.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainé la plus faible IAE et l'architecture ResNet. Tout d'abord, il est observé que les contraintes d'hauteurs d'eau sont respectées pendant la simulation. Par ailleurs, quand les points de consigne changent, des dépassements des hauteurs d'eau sont visibles : Fig.3.23.a à $6000s$ avec h_{r1} et Fig.3.23.b à $9000s$. De plus, il est observé un comportement d'un système à

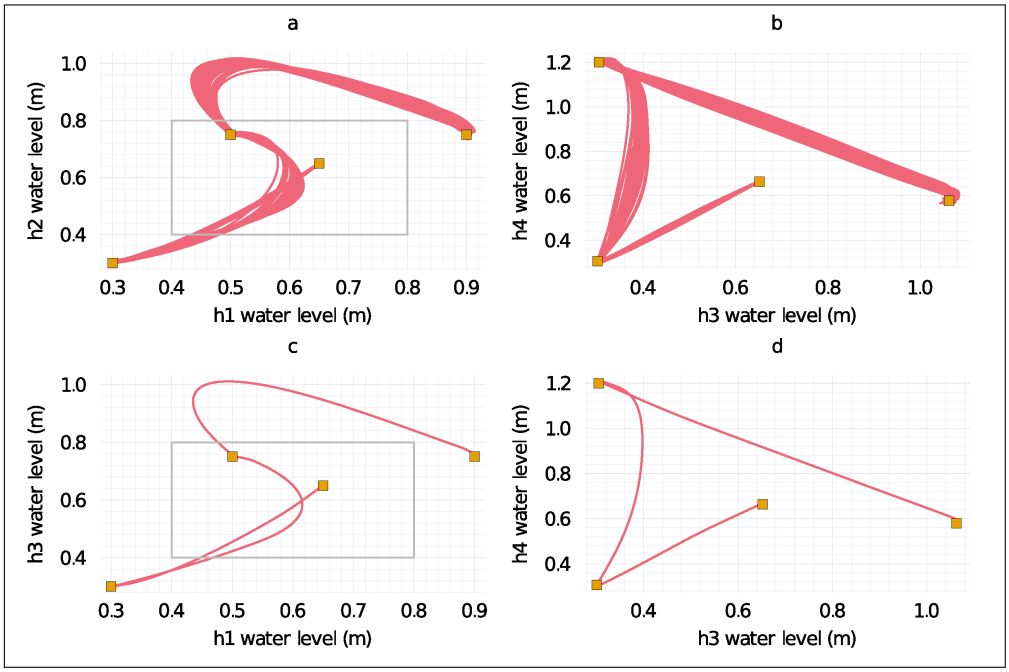

FIGURE 3.22 – Trajectoires de pilotage du système quatres réservoirs avec l'architecture ResNet. Les carrés jaunes indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4.

non minimum de phase avec la Fig.3.23.a à 9000s.

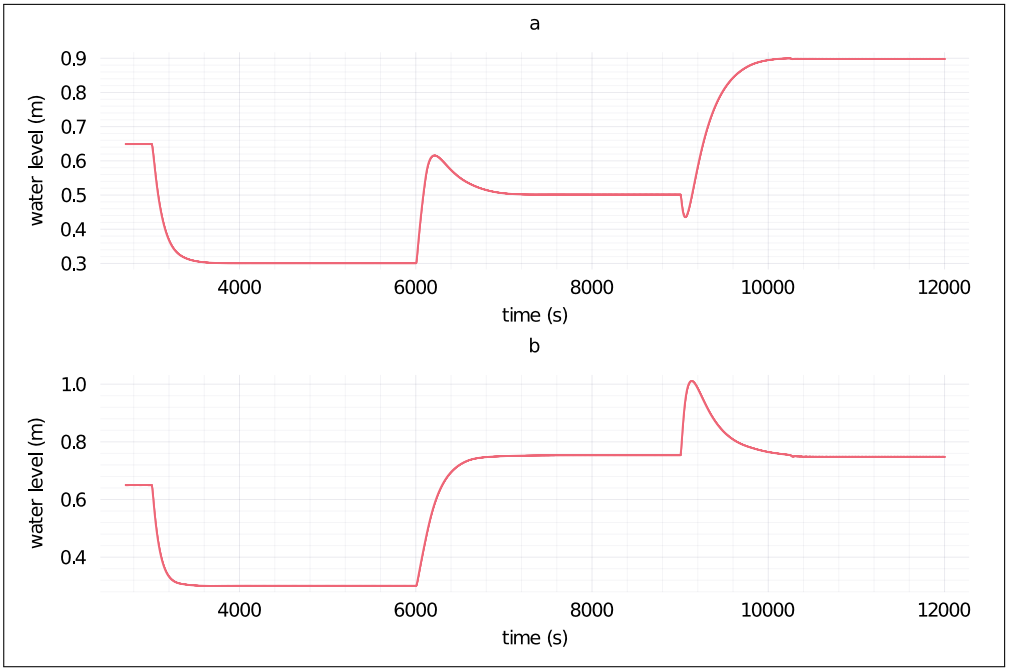

FIGURE 3.23 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture ResNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2.

La Fig.3.24 présente les résultats de la simulation avec la hauteur d'eau du réservoir un (Fig.3.24.a) et deux (Fig.3.24.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainé la plus faible IAE et l'architecture ResNet. En premier lieu, il est observé que les hauteurs d'eau des deux réservoirs restent dans les plages admissibles des hauteurs d'eau. Par ailleurs, il y a un dépassement visible à 6000s sur la hauteur d'eau du réservoir h_{r1}, Fig.3.24.a.

La Fig.3.25 présente les résultats de la simulation avec le débit d'eau de la pompe q_a (Fig.3.25.a) et de la pompe q_b (Fig.3.25.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainés la plus faible IAE et l'architecture ResNet. Nous pouvons observer que le débit des deux pompes reste dans les contraintes admissibles. De plus, les contraintes maximales sont atteintes quand il y a un changement de point de consigne (see Fig.3.25.a à 3000s, 6000s, 9000s et Fig.3.25.b à 3000s, 6000s, 9000s).

La Fig.3.26 présente les résultats des simulations avec les boîtes à moustache de l'IAE selon les états et leur moyenne avec la MPC basée sur le modèle de prédictiction PolyNet. Il est observé en excluant les intrus, que les boîtes sont plus larges que l'évaluation avec l'architecture

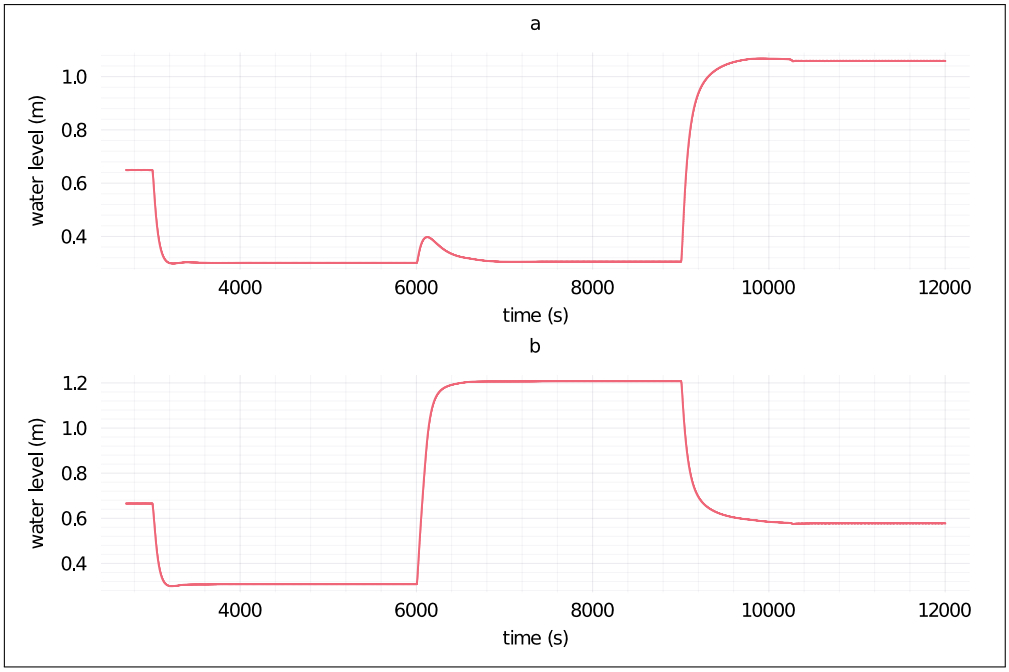

FIGURE 3.24 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture ResNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4.

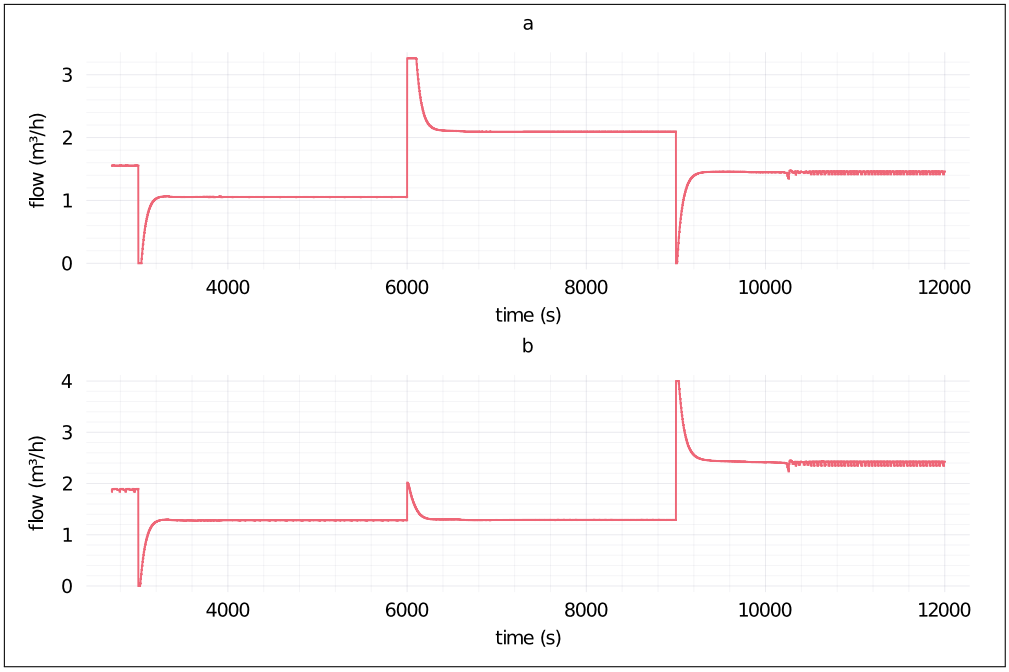

FIGURE 3.25 – Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture ResNet : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b.

Fnn. De ce fait, les différentes simulations réalisées par l'algorithme engendrent un IAE variant plus fortement que les simulations réalisées par le modèle Fnn. Par ailleurs, il est observé que l'IEA sur l'état x_3 engendre un IAE plus faible par rapport aux états, x_1, x_2 et x_4.

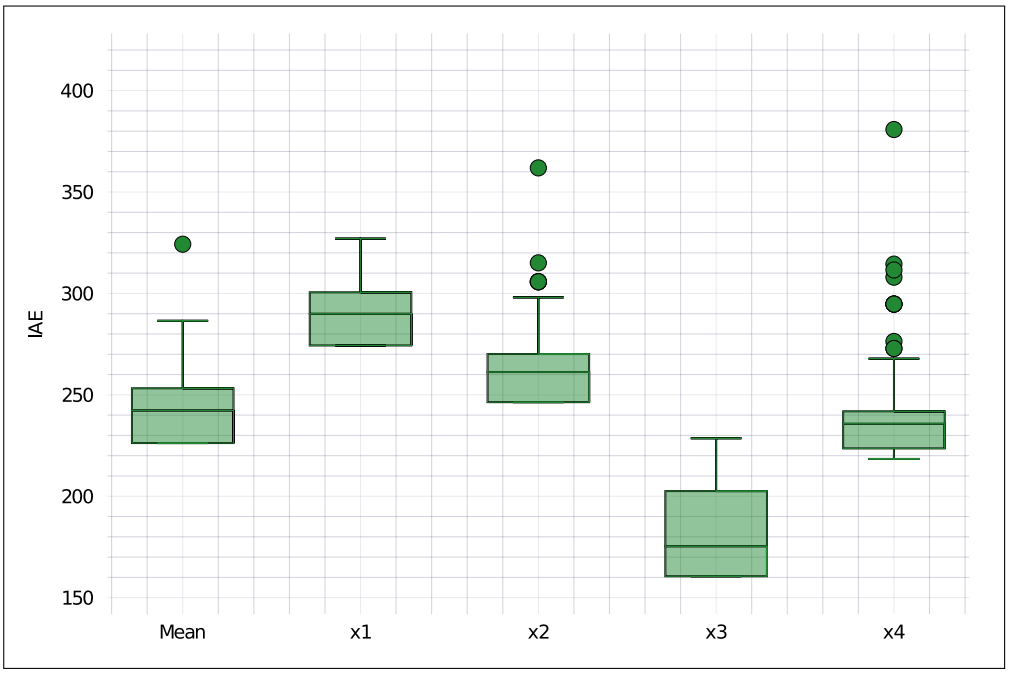

FIGURE 3.26 – boîte à moustache des 150 pilotages du système quatre réservoirs avec l'architecture PolyNet. L'IAE est présentée en fontion des états du système.

La Fig.3.27 présente pour chaque simulation réalisée l'IAE selon les paramètres κ_Q, κ_R et l'horizon N, Fig.3.27.a, Fig.3.27.b et Fig.3.27.c respectivement. En étudiant la Fig.3.27.a, nous pouvons extraire que l'algorithme a évalué plusieurs simulations avec le paramètre κ_Q entre les bornes admissibles. Les résultats des simulations montrent que plusieurs paramètres κ_Q entrainent l'obtention de l'IAE minimum, en effet plusieurs points atteignent un plateau bas. Ensuite, nous pouvons apprécier l'IAE en fonction du paramètre κ_R avec la Fig.3.27.b. Ainsi, entre 0.01 et 0.7, il est observé que plusieurs paramètres ont concouru à trouver le minimum de l'IAE moyen. Néanmoins, utiliser un κ_R entre 0.01 et 0.7 pour ce cas d'étude n'est pas une garantie, car des simulations n'ont pas atteint le minimum observable, de ce fait les paramètres κ_Q et N ont aussi une responsabilité dans la réduction de l'IAE. Finalement, avec Fig.3.27.c nous pouvons observer que l'augmentation de l'horizon jusqu'à des valeurs proches du maximum admissible permet dans certains cas une réduction de l'IAE par rapport à un horizon faible. Cet élément est aussi observé avec les modèles de prédiction Fnn et ResNet.

Fig.3.28 affiche les trajectoires de pilotage des hauteurs d'eau avec l'architecture PolyNet.

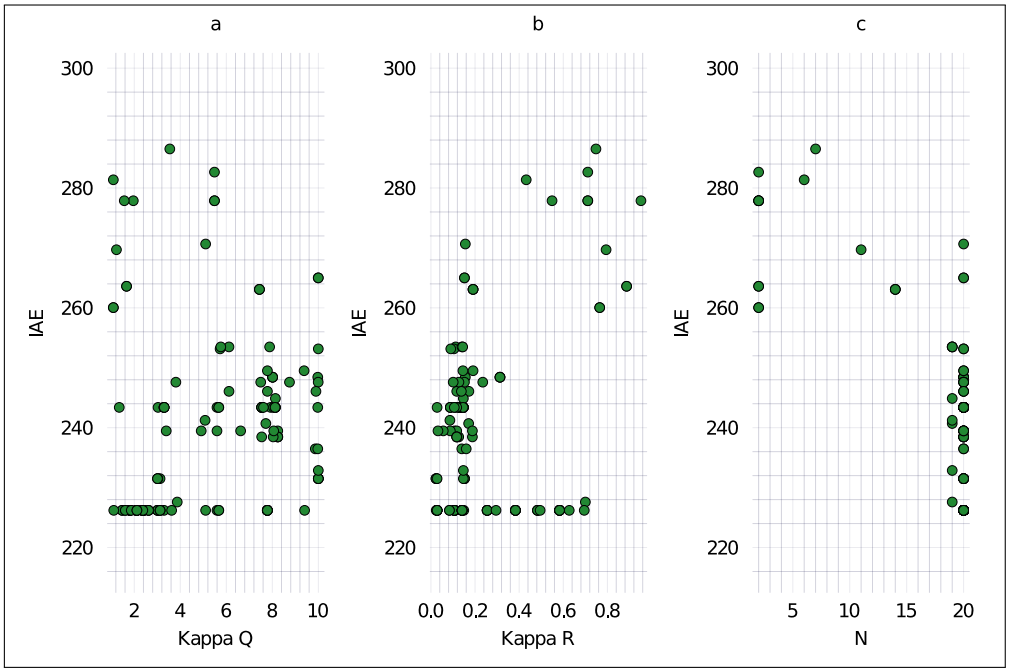

FIGURE 3.27 – IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture PolyNet.

Fig.3.28.a expose les trajectoires de h_2 en fonction de h_1 pour les 150 simulations. Fig.3.28.b présente les trajectoires avec h_4 en fonction de h_3 pour les 150 simulations. Fig.3.28.c présente la trajectoire de h_3 en fonction de h_1 réalisant la plus faible IAE moyen et Fig.3.28.d pour la trajectoire de h_4 en fonction de h_3. Ces figures permettent de mettre en lumière des dépassements éventuels en régime établi et les trajectoires hétérogènes entre les simulations. Ainsi, des trajectoires problématiques sont observées à la Fig.3.28.a et Fig.3.28.b. Ces trajectoires occasionnent des points d'équilibres excentrés par rapport aux points de consigne et des violations des contraintes minimales sur hauteurs d'eau des réservoirs. Concernant, la trajectoire entrainant la plus faible IAE, il n'est pas observé de dépassement aux points de référence, Fig.3.28.c et Fig.3.28.d. Puis, nous pouvons examiner les hauteurs d'eau utilisées pour l'entrainement du prédicteur PolyNet, cette zone est matérialisée avec un rectangle gris sur les Fig.3.28.a et Fig.3.28.c. L'influence des données d'entrainement sur une zone réduite n'est pas visible avec les Fig.3.28.a et Fig.3.28.c.

La Fig.3.29 présente les résultats de la simulation avec la hauteur d'eau du réservoir un (Fig.3.29.a) et deux (Fig.3.29.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainé la plus faible IAE et l'architecture PolyNet. Tout d'abord il est observé que les contraintes de hauteurs d'eau sont respectées pendant la simulation. Par ailleurs, quand

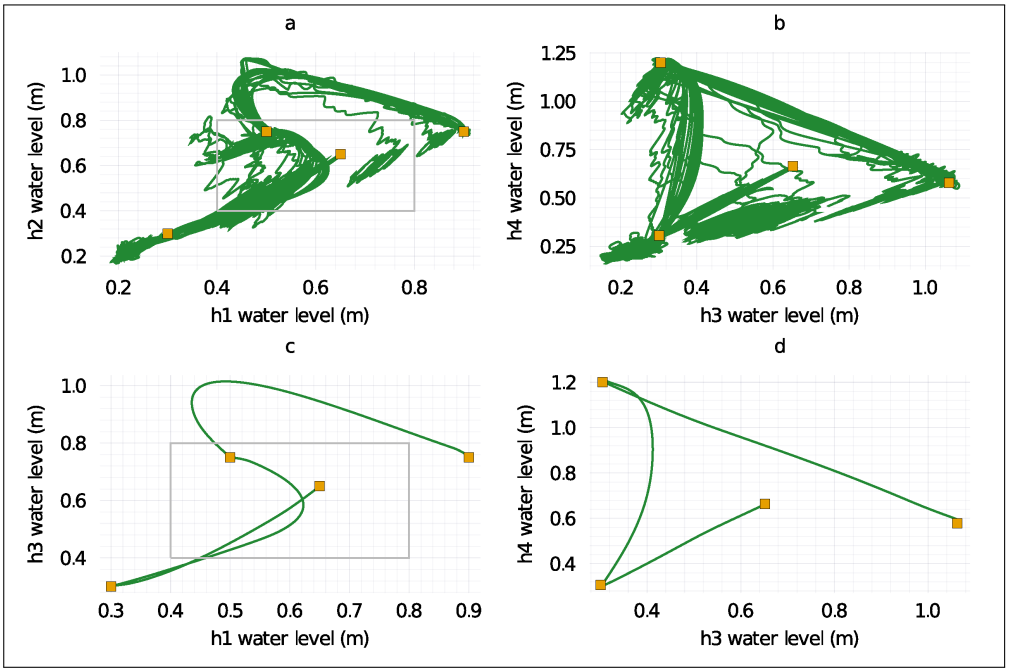

FIGURE 3.28 – Trajectoires de pilotage du système quatres réservoirs avec l'architecture PolyNet. Les carrés jaunes indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4.

les points de consignes changent, des dépassements des hauteurs d'eau sont visibles : Fig.3.29.a à 6000s avec h_{r1} et Fig.3.29.b à 9000s. De plus, il est observé un comportement d'un système à non minimum de phase avec la Fig.3.29.a à 9000s.

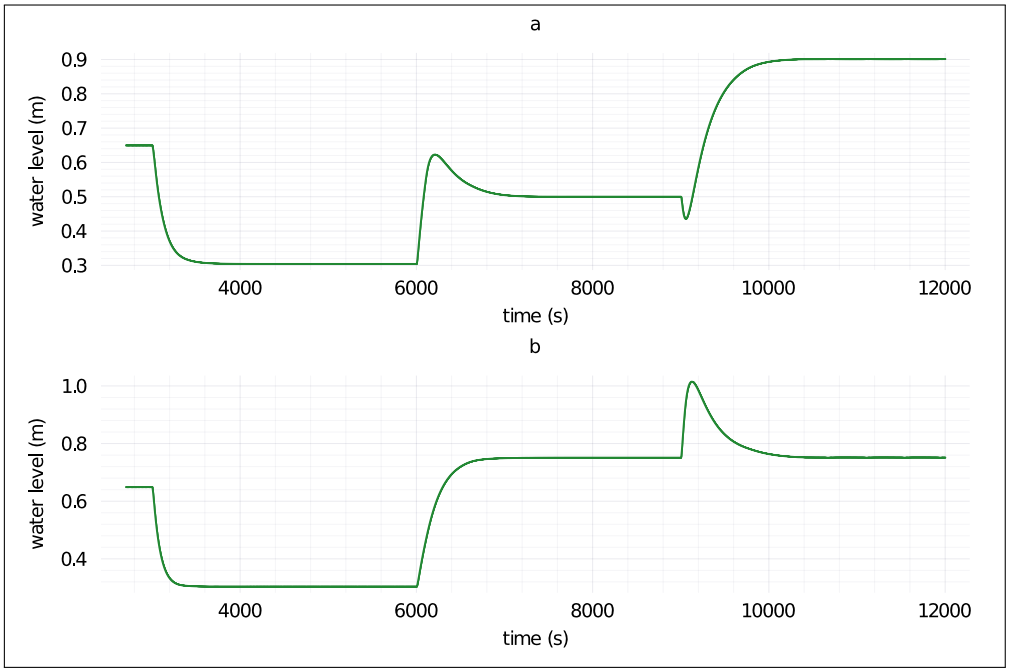

FIGURE 3.29 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture PolyNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2.

La Fig.3.30 présente les résultats de la simulation avec la hauteur d'eau du réservoir un (Fig.3.30.a) et deux (Fig.3.30.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainés la plus IAE et l'architecture PolyNet. En premier lieu, il est observé que les hauteurs d'eau des deux réservoirs restent dans les plages admissibles des hauteurs d'eau. Par ailleurs, il y a un dépassement visible à 6000s sur la hauteur d'eau du réservoir h_{r1}, Fig.3.24.a.

Fig.3.31 présente les résultats de la simulation avec le débit d'eau de la pompe q_a (Fig.3.31.a) et de la pompe q_b (Fig.3.31.b) en fonction du temps. Cette simulation fut réalisée avec les paramètres ayant entrainés la plus faible IAE et l'architecture PolyNet. Nous pouvons observer que le débit des deux pompes reste dans les contraintes admissibles. De plus, les contraintes maximales sont atteintes quand il y a un changement de point de consigne (see Fig.3.31.a à 3000s, 6000s, 9000s et Fig.3.31.b à 3000s, 6000s, 9000s).

La synthèse des paramètres des MPC qui accomplissent les plus faibles IAE moyens pour

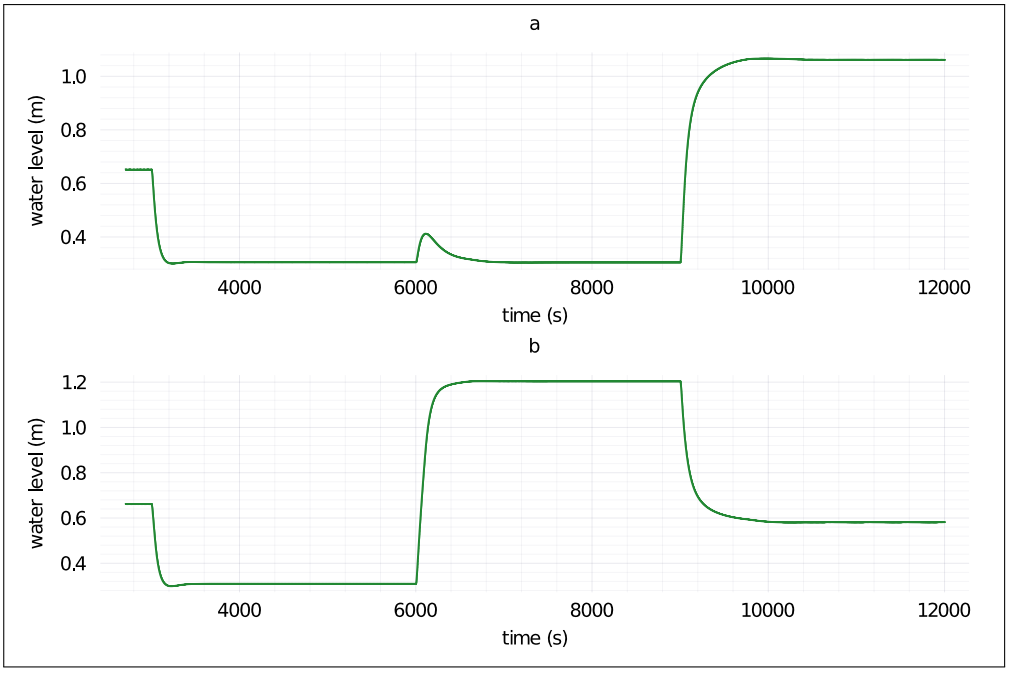

FIGURE 3.30 – Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture PolyNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4.

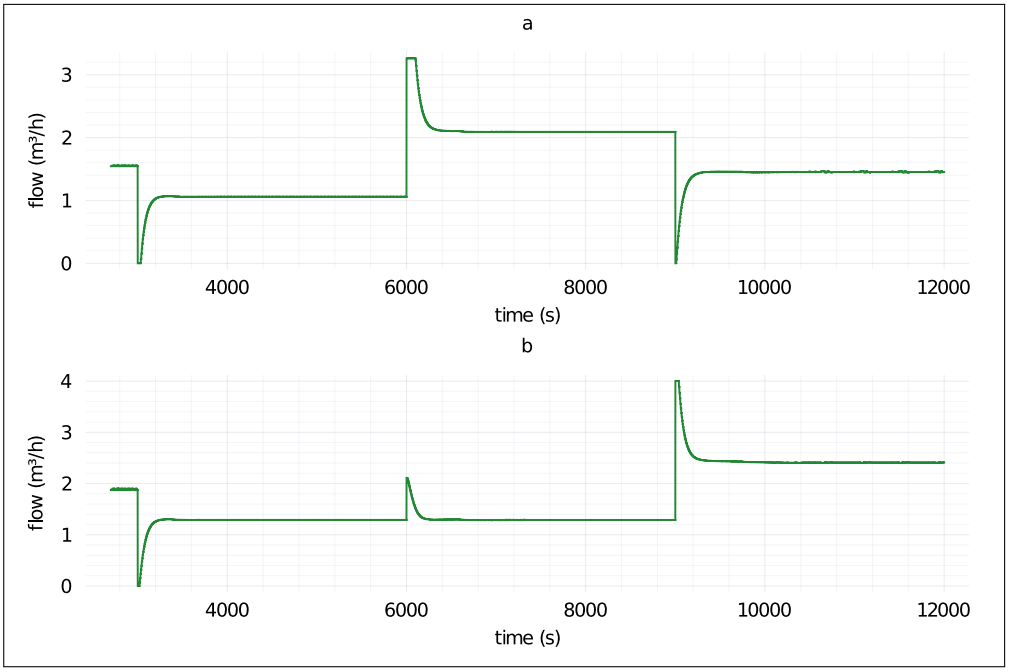

FIGURE 3.31 – Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture PolyNet : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b.

les architectures de prédicteurs Fnn, ResNet et PolyNet est visible à la Tab.3.13. Il est remarqué que l'amélioration de prédiction des états futurs par ResNet et PolyNet par rapport à Fnn (Fig.3.7) a une influence sur l'IAE. En effet, la simulation occasionnant la plus faible IAE avec un prédicteur Fnn est plus importante qu'avec un prédicteur REsNet et PolyNet. Concernant les paramètres de la MPC, un horizon de 19 et 20 occasionnent les simulations avec la plus faible IAE.

TABLE 3.13 – Paramètres de la MPC et de l'IAE.

Valeur	FNN	ResNet	PolyNet
Plus faible IAE : $\int_{t_1}^{t_2} \lvert e(t) \rvert \, dt$	235.19	221.62	226.21
κ_Q	5.58	8.73	9.40
κ_R	0.013	0.71	0.15
N	19	20	20
Couche cachée	1	3	3
Neurones par couche cachée	14	9	8
Fonction d'activation	swish	softsign	swish
Nombre de couche de ResNet ou PolyNet	-	2	6
Numéro du réseau	23	1	24

3.3 Discussion

La synthèse de la méthodologie développée lors de ce travail de recherche est visible à la section 2.1. Les grandes étapes sont : la modélisation fine multiphysique, la génération et récolte des données, la paramétrisation du modèle boîte noire, la synthèse du régulateur et l'évaluation du pilotage. Cette méthodologie a été appliquée sur le système quatre réservoirs pour piloter les hauteurs d'eau. Le pilotage des entrées de commande est réalisé par la commande MPC et les hauteurs d'eau sont régulés aux points de consigne, ainsi la méthodologie a donné satisfaction dans le développement d'un régulateur. Néanmoins, quelques éléments appris et limitant l'application de la méthodologie sont présents.

Tout d'abord, il est à noter que cette méthodologie nécessite une machine de calcul performante et des environnements numériques tels que le langage de programmation julia ou Modelica. Ces éléments peuvent mettre un frein à sa bonne réalisation et engendrer un obstacle. Néanmoins, toutes les briques logiciels nécessaires sont disponibles avec des libraires open source : Modelica, julia, Flux, BlackBoxOptim et Ipopt. En deuxième lieu, il a été observé que les modèles de prédiction utilisés (Fnn, ResNet et PolyNet) concourent à la bonne réalisation

des objectifs de pilotage de la MPC. Néanmoins, les paramètres de la MPC ont une influence. Comme les hyperparamètres d'un réseau de neurones, ils sont liés et il n'est pas possible d'indiquer en avance les bons paramètres. De ce fait, comme pour toute autre technique de commande, une phase de réglage des paramètres de la MPC est nécessaire. Le développement d'une méthodologie systématique, faisant appel à des méta-paramètres en nombre réduit serait une avancée, le problème restant toujours ouvert selon nous. Soulignons toutefois qu'à défaut, la nécessité de faire des évaluations intensives de réglage est rarement possible directement sur un système réel, ce qui valorise le développement du modèle fin comme proposée dans notre méthodologie. En troisième point, il est à noter que les architectures de réseau de neurones dérivés de méthodes d'intégration (ResNet et PolyNet) permettent pour notre cas d'étude une amélioration de la prédiction des états par rapport à Fnn. Cette amélioration contribue à une réduction de l'IAE par rapport à l'architecture Fnn sur les simulations de pilotage. Néanmoins, concernant PolyNet, des difficultés de calcul pour le solveur de résolution de la MPC ont été rencontrés. Ainsi une sélection rigoureuse du prédicteur de la MPC est nécessaire. Quatrièmement, il est indispensable de posséder les consignes du système pour appliquer la méthodologie avec la commande MPC. Ces consignes sont apportées par un étage de planification dans une commande hiérarchique. Une autre possibilité est l'implémentation d'une commande prédictive économique lorsque le coût économique est facilement exploitable. Ce point ne fait pas partie de cette méthodologie et il est considéré comme une poursuite de recherche.

3.4 Conclusion

Ce chapitre a présenté la validation de la méthodologie exposée avec le chapitre 2 de la thèse de recherche. La validation a été réalisée avec un *benchmark* automatique appelé le système quatre réservoirs. Les résultats ont montré que le système est piloté efficacement avec un modèle prédicteur à base de neurones artificiels et un régulateur MPC. Comme on pouvait s'y attendre le réglage de la MPC (architecture, choix des paramètres) a une influence sur les performances obtenues. Le chapitre suivant traitera de l'application de la méthodologie sur un système multi-énergie.

Troisième partie

Application de la méthodologie

Chapitre 4

Pilotage de la serre agricole

Résumé

Les serres agricoles sont présentes dans le monde entier afin d'augmenter la production de légumes, de fruits et de fleurs en s'abstrayant des conditions climatiques extrêmes. Les conditions climatiques de la serre sont régulées et des équipements de production énergétiques sont ajoutés à la serre. Ce système est multi-énergie, car la serre nécessite de la chaleur, de l'électricité et des fertilisants pour les plantes. Actuellement, les principaux équipements énergétiques présents sont : une chaudière, une unité de co-génération (chaleur et électricité), un stockage de la chaleur et pour finir, le fertilisant qui est constitué par l'enrichissement de CO_2.

Le pilotage classique des équipements est réalisé par des règles expertes. L'unité de co-génération est pilotée selon les mois de l'année et la chaudière est surtout utilisée en appoint. Ce régulateur est utilisé, car il existe un tarif avantageux d'achat de l'électricité produite avec bonification pour la valoriser la chaleur. La législation apporte un changement : il est prévu de mettre un terme à ce tarif de rachat. L'électricité sera vendue sur la bourse avec des tarifs de marché fluctuants. Ce chapitre propose un régulateur avancé pour le système de la serre agricole avec la prise en compte de tarif fluctuants de l'électricité. Ce régulateur a été comparé avec le régulateur classique et il a été observé, une baisse de la production de chaleur par la chaudière, une production de chaleur et d'électricité par l'unité de co-génération équivalente, mais dispersée sur l'année, une augmentation de la consommation de fertilisant sur l'année et finalement une baisse de la consommation de gaz naturel.

Une partie des travaux de ce chapitre fait l'objet de la rédaction d'un article pour une soumission prochaine [202].

La présentation de la serriculture et du problème de recherche sont exposés à la section 4.1. Ensuite, l'état de l'art du pilotage d'une serre agricole est visible à la section 4.2. La méthodologie de recherche est ensuite présentée à la section 4.4, avec la modélisation de la serre agricole à la sous-section 4.4.1, la présentation du pilotage classique à la sous-section 4.4.3, du pilotage

avancé à la sous-section 4.4.4 et du plan d'expérience 4.4.5. Ensuite, la présentation de la serre considérée pour l'étude est visible à la section 4.3. Les résultats des simulations sont présentés à la section 4.5. Une discussion des travaux sur la serre agricole est présentée à la section 4.6. La section 4.7 conclut ce chapitre.

4.1 Production agricole sous serre

4.1.1 Serriculture

La serriculture (culture sous serre) permet d'augmenter la production de fruits, de plantes, de fleurs au moyen du pilotage des conditions climatiques et de la fertilisation par rapport à une culture en plein champ. Actuellement, il existe 500 000 hectares de surface agricole mondiale. La répartition est : 210 000 hectares en Europe, 180 000 hectares en Asie, 44 700 en Afrique, 27 000 en péninsule Arabique, Machrek et Iran, 23 000 en Amérique du Nord, 14 000 en Amérique du Sud puis 1 300 en Océanie [203]. Par ailleurs, de nos jours une augmentation de la surface mondiale apparait, excepté en France, Italie, Japon et Israël, Fig.4.1.

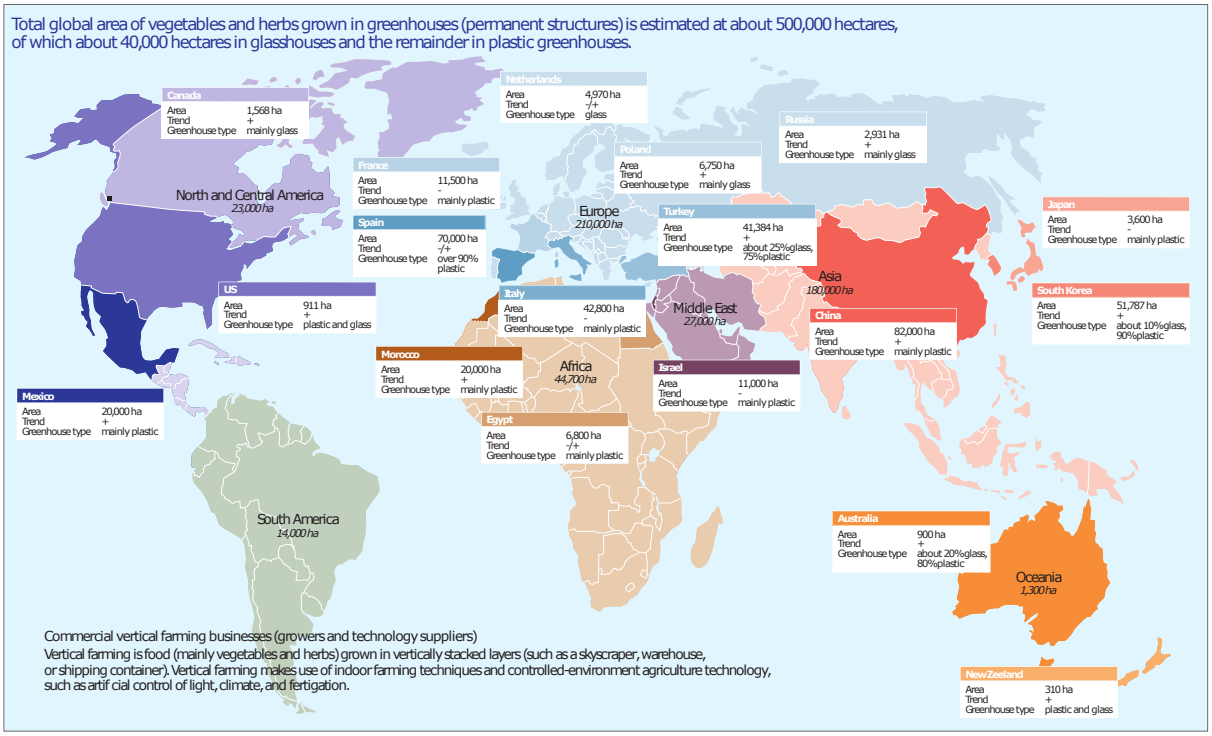

FIGURE 4.1 – Surface mondiale de serre agricole [203], Rabobank.

Les principaux avantages apportés par la serre par rapport à la culture traditionnelle sont :

- L'aléa climatique n'existe plus, occasionnant une augmentation des récoltes et de la qualité des produits (moins de stress pour la plante) ;
- La période de production est augmentée, provoquant une augmentation des récoltes ;
- Une réduction des apports en eau est envisageable par une meilleure utilisation de celle-ci par les plantes ;
- L'utilisation des fongicides et des insecticides est réduite grâce à la protection apportée par la serre.

Cependant, cela occasionne des coûts supplémentaires (construction et exploitation), plus d'énergie est consommée et l'utilisation d'énergie non renouvelable engendre des rejets de CO_2 par rapport aux cultures traditionnelles [204].

Il existe deux types de serres : les serres légères principalement réalisées avec des films plastiques, Fig.4.2 et les serres lourdes en verre. Ces dernières possèdent un nombre important d'équipements supports (chauffage, production énergétique, stockage, éclairage...), Fig. 4.3. Ces serres nécessitent des techniques importantes pour le pilotage des équipements supports. Les cultures réalisées par les serres lourdes sont des cultures à forte valeur ajoutée telles que la tomate ou le concombre réalisé en culture hors-sol (hydroponie/aéroponie).

FIGURE 4.2 – Serre agricole légère.

FIGURE 4.3 – Serre agricole lourde [205], Gestebier, CC BY-SA 4.0, via Wikimedia Commons.

4.1.2 Serre agricole

Les serres lourdes possèdent un nombre important d'équipements et de matériaux pour influencer le climat interne. En effet, les variables climatiques influencent sur la croissance du végétal (température, humidité, luminosité, CO_2 [1]). Leurs gestions nécessitent des équipements spécifiques :

Gestion de la température La gestion de la température est essentielle pour la photosynthèse, la transpiration, la circulation de la sève et les cellules de la plante. Le pilotage de la température relève d'un caractère primordial. Le volume d'air de la serre est protégé du vent et il est réchauffé par le rayonnement du soleil, ces deux phénomènes naturels sont à l'origine de la variation de la température. Concernant la serre lourde, des équipements pour piloter la température en tout temps sont présents. L'hiver elle est réchauffée et l'été elle est refroidie. Les équipements ajoutés sont : un réseau de chauffage, un écran d'ombrage, une ventilation et autre système actif de refroidissement [206].

1. Le périmètre de l'étude est réduit. Il n'est pas considéré tout le champ de travail d'un serriste et intrant de la serre, tel que l'alimentation en eau des plantes, fertilisants, gestion des excroissances, identification des parasites...

Gestion de l'humidité La plante rejette de la vapeur d'eau par la respiration, ce qui entraîne l'augmentation du taux d'humidité. Un taux d'humidité trop important représente un préjudice à la croissance de la plante, car il réduit la photosynthèse et provoque le développement de maladies cryptogamiques. La gestion de l'humidité est réalisée par la ventilation avec l'air extérieur en teneur d'eau réduite ou bien par des systèmes actifs de déshumidification. Les systèmes actifs réduisent l'utilisation de la ventilation et les besoins de chauffages en hiver [206].

Gestion de la luminosité Le rayonnement solaire est essentiel à la plante pour la photosynthèse, pour sa croissance et la production de fruits ou légumes. La longueur d'onde nécessaire aux plantes se situent dans le bleu (400-500 nm) et le rouge (600-700 nm), le vert est réfléchi. Dans les serres lourdes, le verre employé est à forte transmissivité et diffusant pour homogénéiser le rayonnement. De plus, des éclairages artificiels sont ajoutés pour compenser le manque de luminosité en certaines périodes de l'année [206].

Gestion de la concentration de CO_2 La photosynthèse produit le glucose de la plante à partir du CO_2, de H_2O et relâche l'O_2. Le rendement de la plante s'accroît lorsque le CO_2 dépasse la concentration de l'air (environ 400 ppm) jusqu'a 1000 ppm. Ainsi pour augmenter le développement des végétaux, le CO_2 est injecté dans la serre [206].

Nous pouvons en déduire la liste les équipements supports de la serre agricole afin de piloter les variables climatiques (température, humidité, lumière et CO_2) :

- Température : Nécessite une unité de production de chaleur, un réseau de distribution dans la serre, un éventuel stockage de cette chaleur, puis une ventilation et un écran thermique/solaire pour le confort d'été de la plante et la réduction des pertes thermiques la nuit ;
- Humidité : Nécessite une ventilation mécanique et/ou des déshumidificateurs ;
- Luminosité : Nécessite un éclairage artificiel (lampes) ;
- CO_2 : Nécessite une unité de production de CO_2 sur l'exploitation agricole ou bien une unité de stockage avec un injecteur de CO_2.

Un schéma de principe du système serre avec les variables climatiques et les équipements supports sont présentés à Fig.4.4. En outre, en fonction des informations présentées dans la section précédente, les équipements supports de la serre agricole nécessitent aussi une gestion particulière :

Gestion des intrants Les équipements supports sont essentiels au fonctionnement de la serre pour assurer des bonnes conditions climatiques et satisfaire la croissance des végétaux. Une réduction des intrants conduit à une réduction du confort des plantes et de la production agricole.

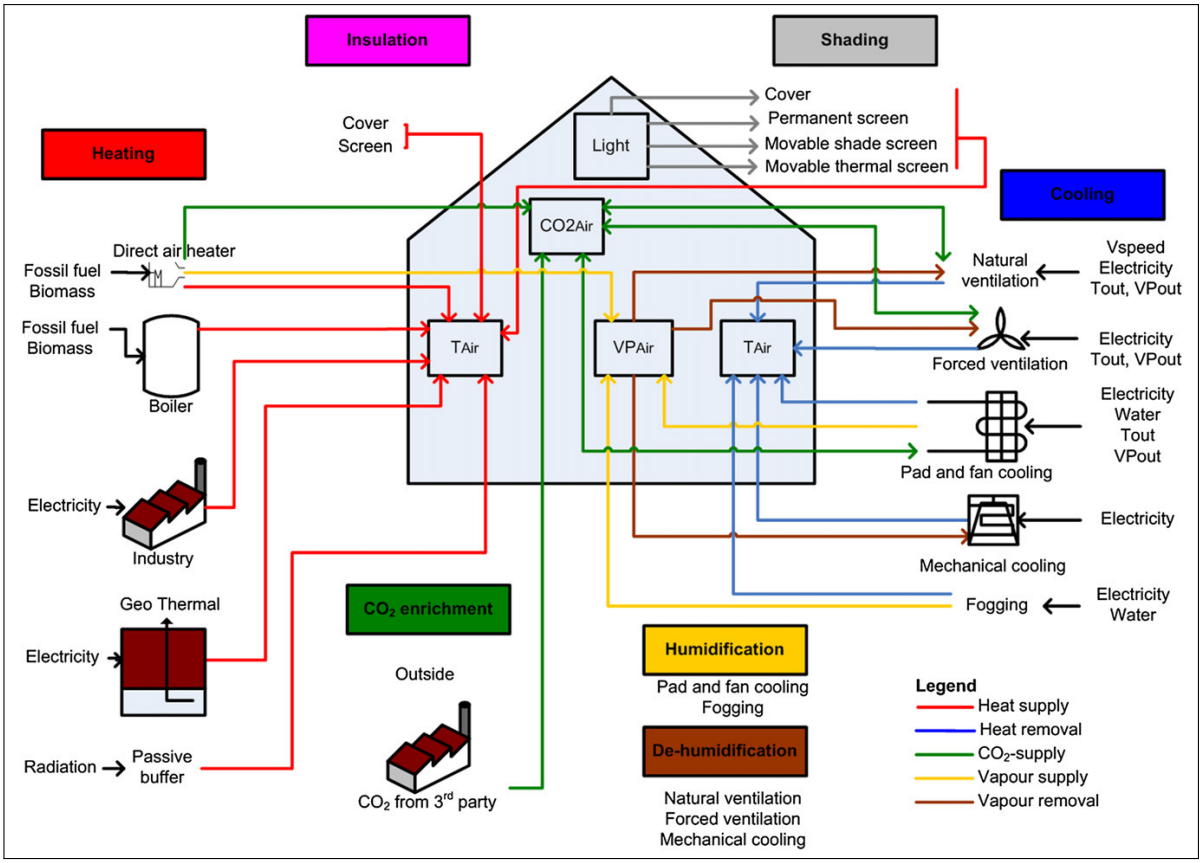

FIGURE 4.4 – Équipements et flux pour gérer les variables climatiques [207].

Finalement, la serre agricole lourde retient notre attention dans ce travail de recherche. En effet, elle possède les éléments de distinction suivants qui motivent notre travail de recherche :

- La serre et les équipements supports sont analysés comme un système multi-énergies. Plusieurs vecteurs énergétiques contribuent à son fonctionnement (chaleur, gaz, électricité) ;
- Le haut niveau de technicité nécessite un pilotage avancé pour la gestion des différents équipements, des variables climatiques tout en visant une réduction des coûts économiques ;
- Le secteur d'activité est fortement présent et dynamique dans la région des Pays de la

Loire et du Grand Ouest ;

— Les contrats de fourniture et de rachat énergétique sont en évolution afin d'être corrélés au tarif du marché. Ces modifications induisent des projections économiques délicates des unités de production énergétiques pour les serristes [2] ;

La section suivante présentera un état de l'art des travaux réalisés pour le pilotage des variables climatiques et des équipements supports de la serre agricole.

4.2 État de l'art : pilotage de la serre

Le pilotage des actionneurs de la serre agricole a été investigué par différentes méthodes et algorithmes de commande. Par exemple, la commande par mode glissant a été employée pour sa capacité à gérer les non-linéarités, le couplage et les perturbations pour piloter l'ambiance de la serre [208]. Les auteurs ont trouvé que la commande par mode glissant est capable de réguler le climat de la serre à la consigne en dépit des non-linéarités et les perturbations. Dans [209], les auteurs ont utilisé la commande par logique floue. Cette dernière à l'avantage de ne pas recourir à l'utilisation d'un modèle physique préétabli, mais formalise et approche le comportement de la personne. Une commande dénommée "sans modèle" est aussi visible dans le document [210], le contrôleur après synthèse est capable de réguler la température avec une faible erreur de consigne. Un régulateur optimal est utilisé avec une fonction coût économique [211], ou bien par un régulateur prédictif [212] pour piloter le système serre agricole. Par ailleurs, les auteurs ont utilisé la méthode de perturbation singulière pour réaliser la séparation des dynamiques rapides et lentes du système [213].

Par ailleurs, la commande hiérarchique a été investiguée [214], la commande est séparée en trois niveaux. Le premier niveau pilote à l'échelle de temps de la minute les fertilisants et le climat de la serre. Le second niveau à l'échelle de temps de la journée adapte les points de consignes et le dernier niveau à l'échelle de temps du mois pilote la croissance de la plante. En outre, une commande hiérarchique à deux niveaux est utilisée dans le document [215] pour le pilotage de la serre. Le premier calcule les points de consigne par une minimisation du coût économique de l'énergie. Le second niveau reçoit les consignes du premier niveau et assure la régulation par des régulateurs PID et bang-bang. Une commande à deux niveaux est aussi examinée dans les travaux de [216], illustré à la Fig.4.5. Le premier niveau réalise une

2. Il existe actuellement un contrat d'achat de l'électricité produite par les serristes. Ce contrat de rachat présente un tarif profitable, car les serristes peuvent valoriser la chaleur fatale pour chauffer la serre. Il est à noter que ce contrat de rachat est amené à disparaitre.

optimisation des profits de la serre agricole et le second niveau réalise le suivi des consignes avec la commande MPC.

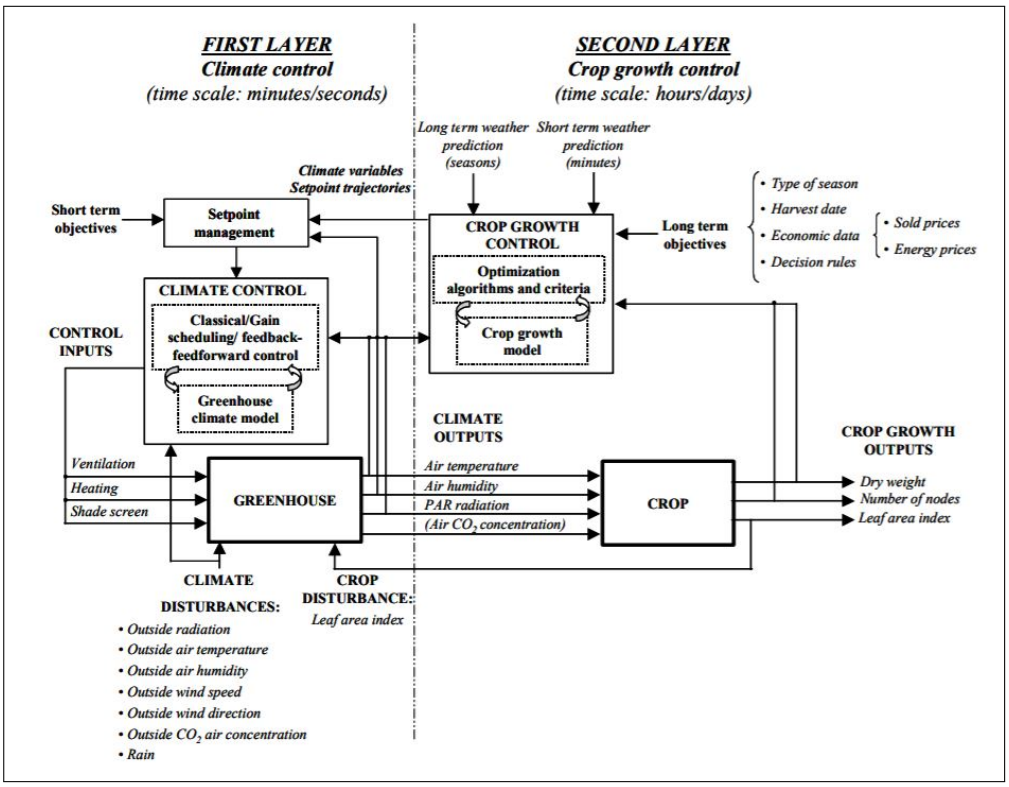

FIGURE 4.5 – Commande hierarchique de la serre agricole de [216] avec les deux niveaux de commande, ©[2003] IEEE.

La commande optimale a été employée dans plusieurs études pour piloter le climat de la serre agricole. La commande optimale permet de piloter la température, l'humidité et la concentration de dioxyde de carbone, tout en minimisant la consommation énergétique [217]. Dans une autre étude, le chauffage et le refroidissement sont pilotés pour réduire la consommation énergétique [218]. Le formalisme de commande optimale par Hamilton Jacobi Bellman (HJB) est aussi utilisé dans [219]. En outre, une optimisation multiobjective est investiguée dans [214]. Elle vise la maximisation de la production végétale et de la qualité des fruits avec la minimisation de la consommation en eau. La commande avancée MPC est également considérée dans [220], pour sa capacité à prendre en compte des contraintes, ainsi que le caractère non linéaire et multivariable du système. Dans [221], il est montré qu'une commande MPC non linéaire apportait de meilleures performances que le régulateur PID en regard de la régulation de température. Une version robuste de la MPC pour prendre en compte les incertitudes est aussi abordée, les résultats indiquent que le régulateur régule la température en dépit des incertitudes

avec un algorithme d'optimisation min-max [222].

Malgré les études nombreuses pour réguler le climat de la serre agricole, peu d'études ont abordé le pilotage du système énergétique support à la serre agricole. En particulier, la considération du système support multi-énergies n'est pas étudiée afin d'augmenter l'efficacité de conversion, l'amélioration de la flexibilité et l'utilisation optimale des ressources énergétiques.

Un outil développé sous Matlab intitulé ODEHubs permet de configurer un hub énergétique appliqué à la serre agricole et support énergétiques. Les équipements sont, des panneaux photovoltaïques, un réseau électrique, un réseau en eau, des stockages, un chauffage par propane, une chaudière biomasse, puis la serre agricole est considérée comme la charge[223]. Cependant, les équipements sont modélisés par un modèle linéaire afin de conserver une optimisation linéaire. Par ailleurs, le travail réalisé dans [224] présente une serre agricole dans un microréseau avec intelligence pour piloter efficacement l'énergie, augmenter la disponibilité, la flexibilité et l'efficacité. Le pilotage est réalisé par une MPC en prenant en compte, des éoliennes, des panneaux photovoltaïques, des stockages par batteries, une connexion au réseau et la chaleur par une co-génératrice. Les auteurs ont développé leurs travaux en prenant en compte un réseau de plusieurs serres interconnectées par un microréseau [225]. Dans les deux travaux [224], [225], la modélisation de la charge de la serre est réalisée par un modèle nodal résistance-capacité et le processus de croissance de la plante n'est pas prise en compte. De plus, l'étude avec des systèmes énergétiques support constitué d'éoliennes et de photovoltaïques ne reflète pas la réalité des exploitations maraîchères. En effet, les surfaces agricoles sont conservées pour la production végétale à la place de la production énergétique par des panneaux solaires.

Un système énergétique support plus réaliste est visible dans le document [226] avec un pilotage optimal des équipements, chaudière, co-génératrice, et deux stockages thermiques. L'optimisation a pour objectif de minimiser le coût du gaz naturel et de l'électricité tout en prenant en compte les contraintes du système. De plus, les auteurs ont comparé le pilotage optimal avec le pilotage classique et ils ont obtenu une augmentation du bénéfice de 20%. Une réduction du coût économique de 29% sur une année est obtenu avec l'addition d'une pompe à chaleur, d'un stockage énergétique de temps long et de tours de refroidissement [227]. Néanmoins, les deux études [226], [227] ne prennent pas en compte les capacités de la MPC avec un horizon glissant, et les systèmes énergétiques sont considérées de manière simplifiée en faisant l'hypothèse d'un rendement constant.

En outre, les travaux les plus corrélés à notre problématique [223]-[227] reposent sur une modélisation physique simplifiée des systèmes serre agricole et supports énergétiques. Dans le but d'observer l'efficacité de conversion, la flexibilité des systèmes et le déploiement optimal des

ressources énergétiques, il est envisagé d'améliorer la précision de la modélisation du système multi-énergies et d'appliquer la méthodologie de pilotage présenté dans le chapitre 2. La démarche scientifique remployée pour ce cas d'étude est dans un premier temps de réaliser une modélisation détaillée multiphysique du système multi-énergie avec un programme Modelica. Ce programme est basé sur les librairies Greenhouse [92] et TransiEnt [94]. Ensuite, dans un deuxième temps la méthodologie développée lors de ce travail de thèse n'est pas appliquée de suite. Une première simulation est réalisée pour évaluer le pilotage actuel du système avec un régulateur basé sur des règles expertes. Puis, une seconde simulation est effectuée avec un régulateur hiérarchique à deux niveaux. Le premier niveau réalise l'optimisation du coût économique et le second niveau exécute le pilotage des équipements. Ce second niveau reprend la méthodologie et l'algorithme de pilotage développé dans ce travail de recherche (chapitre 2).

4.3 Présentation de la serre agricole de l'étude

La serre agricole considérée pour l'étude est synthétique. Elle représente une serre des Pays de la Loire sans que les paramètres soient parfaitement équivalents. Ainsi, la serre numérique est représentative des serres lourdes du département à l'échelle réduite. Cette approche permet de simplifier l'étude et de proposer une évolution éventuelle dans le cas d'une application pour une exploitation et d'une poursuite de l'étude de recherche engagée dans cette thèse. En effet, pour une même exploitation maraîchère ils existent plusieurs serres et plusieurs cultures végétales. Idem pour les équipements supports, il peut y avoir plusieurs équipements supports du même type (ex. chaudière) avec des paramètres distincts.

Le système serre est considéré comme un système multi-énergie. En effet, plusieurs vecteurs sont présents sur l'exploitation. Ces vecteurs sont : le gaz naturel, la chaleur, l'électricité et un fertilisant par le CO_2. Ces vecteurs résultent des équipements sélectionnés et des demandes énergétiques :

- Demandes énergétiques par la serre agricole, chaleur, électricité et fertilisant par le CO_2 ;
- Distribution de la chaleur à la serre par deux réseaux de chaleur : haute température et basse température ;
- Stockage de la chaleur (seulement en haute température) ;
- Chaudière consommant du gaz naturel pour produire de la chaleur avec une valorisation du CO_2 pour l'injection dans la serre agricole ;
- Unité de co-génération produisant de l'électricité et de la chaleur à partir du gaz naturel ;
- Connexion à deux réseaux énergétiques, électricité et gaz naturel puis connexion à un

TABLE 4.1 – Paramètres synthétiques du système serre agricole.

Description	Symbole	Paramètres	Paramètres par hectare
Surface	A_{gh}	60 000m^2	6ha
Hauteur	h_{gh}	4m	-
Plante	-	Tomate	-
Puissance des lampes	E_{gh}	$1.8 \times 10^6 W$	$0.3 \times 10^6 W/ha$
Puissance de la chaudière	H_{bl}	$6.6 \times 10^6 W$	$1.1 \times 10^6 W/ha$
Puissance thermique de la co-génératrice	H_{chp}	$4.4 \times 10^6 W$	$0.73 \times 10^6 W/ha$
Puissance électrique de la co-génératrice	E_{chp}	$4.4 \times 10^6 W$	$0.73 \times 10^6 W/ha$
Volume stockage thermique	V_{tes}	1500m^3	250m^3/ha
Température stockage thermique SOC = 1	-	90°C	-
Température stockage thermique SOC = 0	-	55°C	-

réseau de distribution de CO_2.

Finalement, les différents équipements multi-énergies de l'étude sont : un stockage de la chaleur, une chaudière, enfin une unité de co-génération. Ainsi, pour conserver l'étude synthétique il n'est pas considéré des équipements énergétiques exotiques que l'on peut retrouver sur des exploitations, mais qui ne sont représentatifs (pompe à chaleur géothermie, récupération du CO_2 de l'unité de co-génération, panneaux solaires photovoltaïques...). Par ailleurs, les paramètres des équipements énergétiques de la serre synthétique sont visibles à la Tab.4.1.

4.4 Méthodologie

4.4.1 Modélisation de la serre agricole

La modélisation de la serre agricole et la mise en forme présentées dans les sections suivantes découlent en partie du travail de modélisation de systèmes multi-énergies de [83].

Modélisation boîte grise de la serre agricole

Le comportement de la serre peut être modélisé par un système dynamique non linéaire, couplé et complexe [228]. Afin de simplifier la modélisation, il est considéré une approche nodale par boîte grise avec les équations différentielles ordinaires. Les différentes variables considérées sont : la température de l'air de la serre T_{air}, la concentration en eau de l'air de la serre : X_{air}, la concentration en CO_2 de l'air de la serre $CO_{2_{air}}$, le stockage thermique de la chaleur T_{tes} et la production de chaleur d'une unité sur l'exploitation agricole T_m. Pour ce faire,

un bilan de puissance thermique, un bilan d'humidité et un bilan de CO_2 sont présentés.

Bilan de puissance thermique Le bilan de puissance thermique de la serre agricole est [204] :

$$\boxed{C_{air}\dot{T}_{air} = Q_{gh} + Q_{sol} + Q_{lamp} - Q_{convd} - Q_{rad} - Q_{trans} - Q_{ventil}} \tag{4.1}$$

avec C_{air} la capacité thermique de l'air (JK^{-1}), $\dot{T}_{air}$ la dérivée de la température de l'air $^{\circ}Cs^{-1}$, Q_{gh} la puissance de chauffage de la serre (W), Q_{sol} la puissance du rayonnement solaire (W), Q_{lamp} la puissance thermique dégagée par les luminaires (W), Q_{convd} la puissance thermique de perte par la convection et la conduction de l'enveloppe en verre (W), Q_{rad} la puissance de perte radiative (W), Q_{trans} la puissance thermique absorbée par la plante lors de sa transpiration (W) et Q_{ventil} la perte thermique par la ventilation (W). Pour une simplification de la lecture, les variables Q_{gh}, Q_{sol}, Q_{lamp}, Q_{convd}, Q_{rad}, Q_{trans}, et Q_{ventil} sont présentées à l'Annexe F.

Bilan d'humidité Le bilan d'humidité de la serre agricole est [218] :

$$\boxed{h\dot{\mathrm{X}}_{air} = \phi_{\mathrm{X}_{trans}} - \phi_{\mathrm{X}_{cond}} - \phi_{\mathrm{X}_{desh}} - \phi_{\mathrm{X}_{vent}}} \tag{4.2}$$

avec h est la hauteur de la serre agricole (m), $\dot{\mathrm{X}}_{air}$ la dérivée de la concentration en eau de l'air de la serre $gm^{-3}s^{-1}$, $\phi_{\mathrm{X}_{trans}}$ la vapeur d'eau générée par la transpiration des plantes ($gm^{-2}s^{-1}$), $\phi_{\mathrm{X}_{cond}}$ la condensation sur les vitres de la serre agricole ($gm^{-2}s^{-1}$), $\phi_{\mathrm{X}_{desh}}$ la condensation réalisée par le déshumidificateur ($gm^{-2}s^{-1}$) et $\phi_{\mathrm{X}_{vent}}$ le flux de vapeur résultant de l'action de la ventilation ($gm^{-2}s^{-1}$). Pour une simplification de la lecture, les variables $\phi_{\mathrm{X}_{trans}}$, $\phi_{\mathrm{X}_{couv}}$, $\phi_{\mathrm{X}_{desh}}$ et $\phi_{\mathrm{X}_{vent}}$ sont présentées à l'Annexe F.

Bilan du CO_2 Le bilan du CO_2 de la serre agricole est [217] :

$$\boxed{h\dot{C}O_{2_{air}} = \phi_{CO_{2_{inj}}} - \phi_{CO_{2_{ass}}} - \phi_{CO_{2_{vent}}}} \tag{4.3}$$

avec h la hauteur de la serre agricole (m), $\dot{C}O_{2_{air}}$ la dérivée de la concentration en CO_2 de l'air de la serre $gm^{-3}s^{-1}$, $\phi_{CO_{2_{inj}}}$ l'injection de CO_2 dans la serre agricole ($gm^{-2}s^{-1}$), $\phi_{CO_{2_{ass}}}$ le CO_2 assimilé par les plantes ($gm^{-2}s^{-1}$) et $\phi_{CO_{2_{vent}}}$ l'échange de CO_2 entre la serre et l'air extérieur occasionné par la ventilation. Pour une simplification de la lecture, les variables $\phi_{CO_{2_{inj}}}$, $\phi_{CO_{2_{ass}}}$ et $\phi_{CO_{2_{vent}}}$ sont présentées à l'Annexe F.

Stockage thermique Les serres agricoles possèdent des stockages de chaleur. Pour une modélisation simplifiée avec un mélange homogène de la température de l'eau, un noeud du stockage thermique, le modèle de bilan de puissance est [229] :

$$\boxed{C_{tes}\dot{T}_{tes} = Q_c - Q_d - Q_{pert}} \tag{4.4}$$

avec C_{tes} la capacité thermique du fluide du stockage de la chaleur JK^{-1}, $\dot{T}_{tes}$ la dérivée de la température de stockage $^{\circ}Cs^{-1}$, Q_c la puissance de chauffage (W), Q_d la puissance soutirée (W) et Q_{pert} la puissance perdue (W). Pour une simplification de la lecture, les variables Q_c, Q_d, Q_{pert} ainsi que le cas d'un stockage thermique avec la stratification sont présentées à l'Annexe F.

Unité de production de chaleur Tout d'abord, il est considéré que la serre agricole possède au moins une unité de production pour la chaleur. Nous nous limiterons à la production de chaleur à partir d'un combustible fossile. Le bilan énergétique d'une chaudière simplifiée est [230] est :

$$\boxed{C_{boil}\dot{T}_m = Q_{in} - Q_{go} - Q_e - Q_{pert}} \tag{4.5}$$

avec C_{boil} la capacité thermique de la chaudière (masse de l'échangeur) (Jk^{-1}), $\dot{T}_m$ la dérivée de la température de l'échangeur $^{\circ}Cs^{-1}$, Q_{in} la puissance résultant de la combustion (W), Q_{go} la puissance perdue dans les gaz d'échappement (W), Q_e la puissance exportée par le caloporteur de la chaudière (W) et Q_{pert} la puissance perdue dans l'atmosphère (W). Pour une simplification de la lecture, les variables Q_{in}, Q_{go}, Q_e et Q_{pert} sont présentées à l'Annexe F. D'autres unités de production énergétique peuvent être envisagées pour la serre agricole. Par exemple, une unité de co-génération produisant de la chaleur et de l'électricité, une pompe à chaleur géothermique ou une pile à combustible. Par ailleurs, un schéma de principe du système serre avec les variables climatiques et les flux sont visibles à la Fig.4.6.

Nous pouvons questionner ces modèles boîte grise de la serre agricole avec les éléments suivants :

- Une serre agricole possède plusieurs chapelles sur plusieurs hectares. Modéliser celle-ci avec un noeud est très restrictif (température, humidité et CO_2 homogènes) ;
- La température de la couverture de la serre agricole est nécessaire pour le modèle du bilan d'humidité pour calculer la condensation sur les vitres alors que le modèle thermique ne prend en compte cette variable ;
- Il n'est pas pris en compte l'effet du rideau thermique sur la modélisation boîte grise,

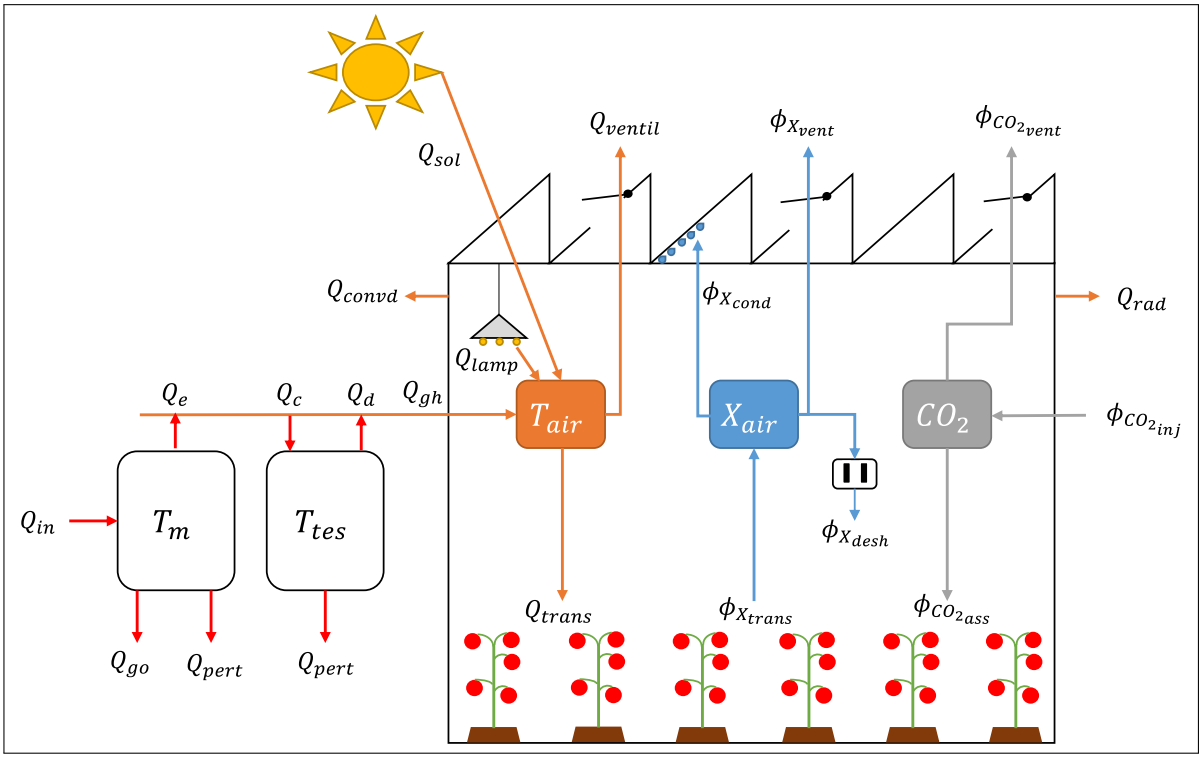

FIGURE 4.6 – Illustration des bilans des variables climatiques avec la serre agricole et des équipements supports.

celui-ci à un impact important sur le bilan thermique de la serre ;
— Le réseau de distribution de chauffage n'est pas pris en compte dans le bilan thermique.

Il existe une librairie Modelica qui modélise finement la serre et la production agricole de tomate [92]. Elle tente de combler les lacunes des équations physiques présentées dans cette section :

— Le modèle thermique de la serre agricole est discrétisé en plusieurs noeuds ;
— L'influence thermique de couverture en verre de la serre est étudiée dans le bilan thermique ;
— Un écran thermique est pris en compte dans le bilan de la température, d'humidité et de CO_2 ;
— Un réseau de distribution de chauffage est modélisé ;
— Un indice de surface foliaire dynamique est pris en compte.

Par ailleurs, il existe une librairie Modelica pour modéliser finement les équipements énergétiques, TransiEnt [94]. Elle permet d'utiliser des modèles détaillés de stockage de la chaleur, de chaudière, d'unité de co-génération et aussi le transport de la chaleur.

Modélisation multiphysique de la serre agricole

Le modèle détaillé supportant la serre synthétique est décrit et implémenté au travers du langage Modelica [88] et du logiciel Dymola [95]. Les librairies utilisées sont Greenhouses [92] pour la modélisation de la serre et TransiEnt [94] pour la modélisation de la chaudière, de l'unité de co-génération, du stockage thermique, des pompes et des vannes. Par ailleurs, la librairie TransiEnt a pour base la librairie Clara [231]. Des apports et des modifications ont été réalisés pour disposer de deux réseaux de chaleur, à basse et haute températures respectivement. De même, les régulateurs bas niveaux ont été implémentés (p. ex. PID pour la régulation de température des réseaux de chaleur). Les différents paramètres visibles à la Tab.4.1 sont renseignés dans le modèle, puis des connecteurs d'entrées et de sorties sont ajoutés pour l'interface avec la FMU et les fichiers météorologiques. Le schéma Modelica du système complet est visible à la Fig.4.7, puis le schéma Modelica de la serre est visible à la Fig.4.8.

4.4.2 Pilotage des variables climatiques

Le pilotage des variables climatiques de la serre agricole par un régulateur avancé n'est pas investigué dans cette étude. Néanmoins, ces variables sont pilotées dans notre environnement de simulation. Afin, de les piloter, des régulateurs PID et *rule-based* sont utilisés. Ces régulateurs

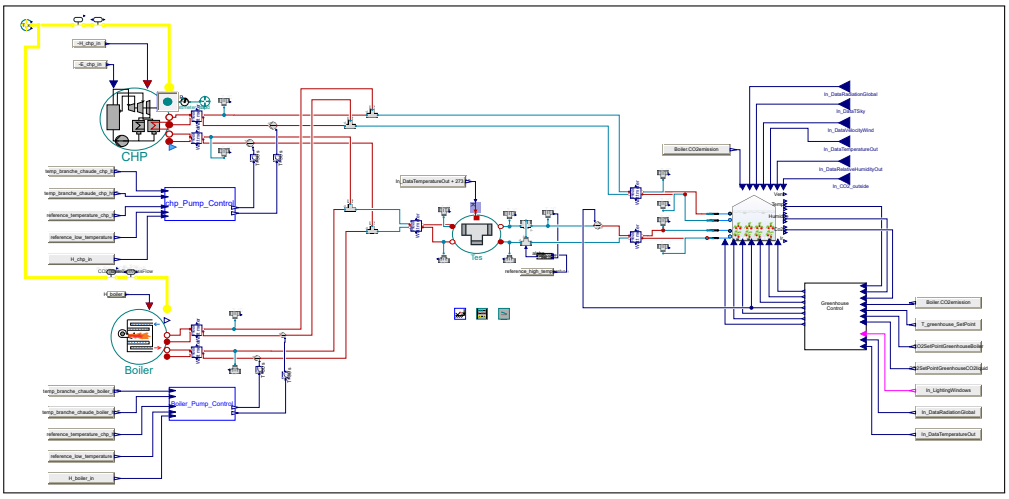

FIGURE 4.7 – Schéma Modelica du processus serre agricole avec le système multi-énergies et la serre.

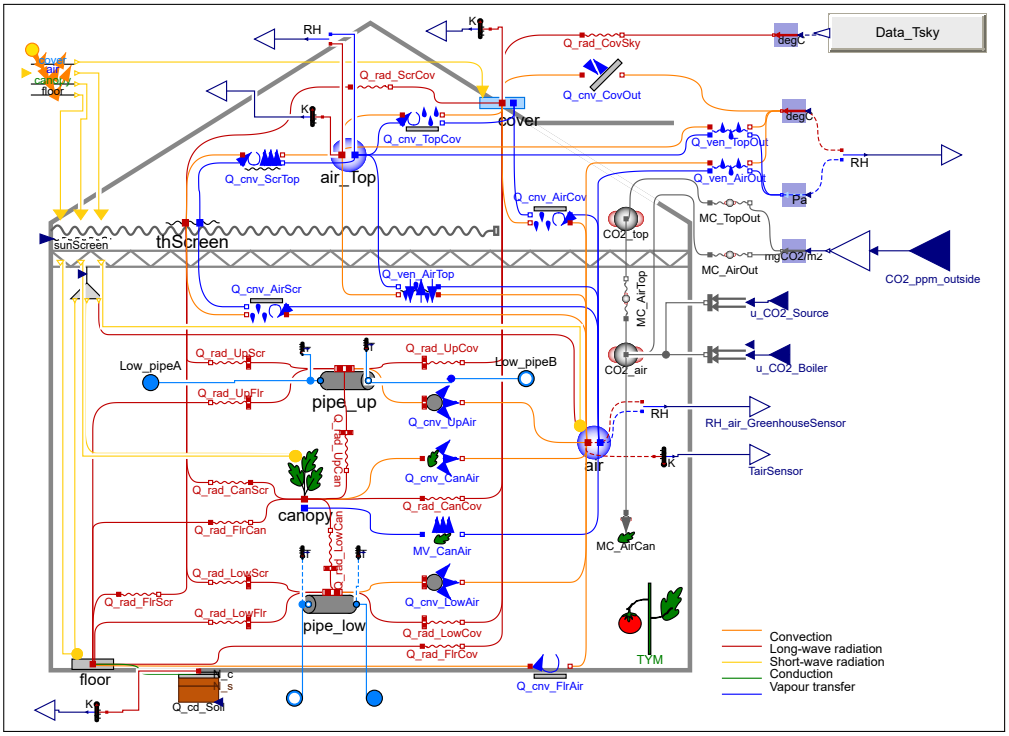

FIGURE 4.8 – Schéma Modelica de la serre agricole tiré de [92] et adapté pour le cas d'étude.

sont extraits de la librairie Greenhouse [92]. Les différents régulateurs remployés sont :

- Pilotage de la température de la serre : Un régulateur PID actionne le débit d'une pompe du circuit de chaleur afin de chauffer la serre lorsque cela est nécessaire. Les points de consignes renseignés sont : $22°C$ entre 6 heures et 21 heures et $20°C$ le reste de la journée.
- Pilotage de l'humidité de la serre : Un régulateur PID actionne les ouvertures des fenêtres de la serre agricole lorsque la concentration d'humidité dépasse les 85% ;
- Pilotage du rayonnement : Le rayonnement artificiel de la serre est piloté par un régulateur de règles expertes. Lorsqu'il y a autorisation d'actionner les lampes (entre 6 et 21 heures) celles-ci sont pilotées par une hystérésis avec seuil haut de $120\,Wm^{-2}$ et bas de $40\,Wm^{-2}$.
- Pilotage de la concentration en CO_2 : Le pilotage de la concentration de CO_2 est réalisé par un régulateur PID. Ce régulateur actionne un injecteur. La consigne de concentration de CO_2 est de $1000ppm$ entre 6 et 21 heures. Le reste de la journée, il n'y a pas de consigne sur la concentration de CO_2.
- Pilotage du confort thermique de la tomate : L'écran thermique est piloté par un régulateur basé sur des règles expertes. La consigne de fermeture ou d'ouverture est réglée sur le seuil de rayonnement de $35\,Wm^{-2}$. L'écran solaire est piloté selon la température et le rayonnement solaire par deux hystérésis connectées sur une porte *ET*. Il y a autorisation par l'hystérésis sur la température avec seuil bas de $27°C$ et seuil haut de $30°C$ et l'hystérie sur le rayonnment solaire avec un seuil bas de $900\,Wm^{-2}$ et seuil haut de $1000\,Wm^{-2}$. Finalement, pour protéger la tomate d'une température trop importante, il y a ouverture des fenêtres si la température de la serre dépasse de $2°C$ la température consigne.

Une illustration des régulateurs pilotant les variables climatiques du processus serre est visible à la Fig.4.9.

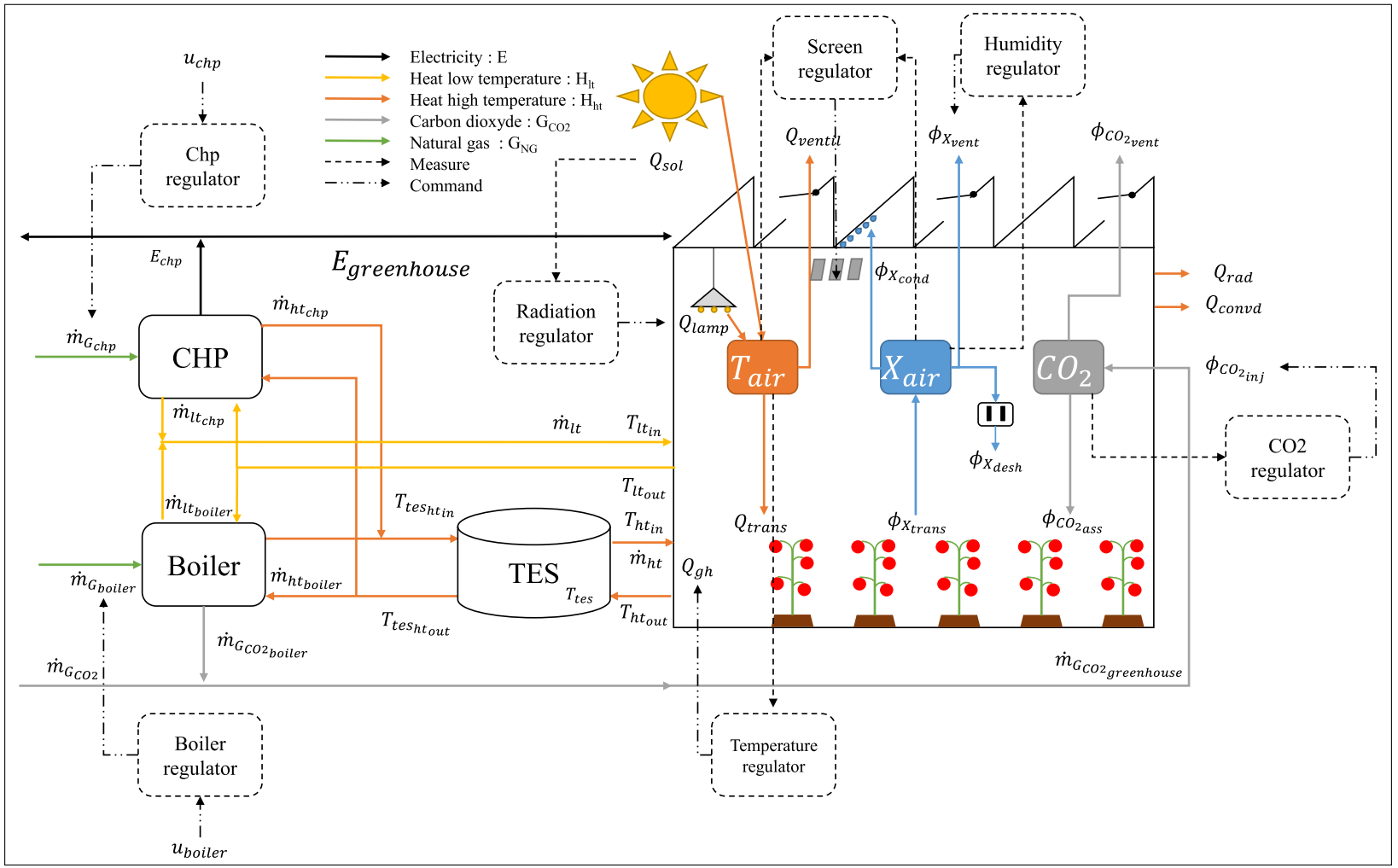

FIGURE 4.9 – Schéma de la serre agricole avec les variables climatiques, les équipements supports et les régulateurs.

4.4.3 Pilotage classique des équipements supports

Le pilotage classique des équipements supports est dérivé des méthodes de pilotage actuelles. Ces méthodes ne s'appuient pas sur des algorithmes avancés, mais sur des habitudes et des volontés de réduction des coûts et de valorisation des intrants (électricité, gaz naturel, CO_2). En effet, actuellement il existe des contrats de rachat sur l'électricité produite. Les équipements à piloter sont : la chaudière, l'unité de co-génération et le stockage thermique. Cependant, le stockage thermique n'est pas piloté directement. En effet, il ne possède pas d'entrée de commande. En outre, des contraintes inhérentes au système sont présentes : les contraintes de puissance sur les équipements et les différents tarifs de l'électricité, du gaz naturel et du CO_2.

Le pilotage des équipements supports est réalisé unitairement sans considération des couplages éventuels entre vecteurs énergétiques. Les objectifs de commande sont focalisés sur l'unité de co-génération dans un premier temps puis la chaudière accessoirement :

- Unité de co-génération : l'unité de co-génération est pilotée selon les mois de l'année avec un fonctionnement à pleine charge de début novembre jusqu'à fin mars. Ensuite, la puissance est nulle le reste de l'année. Ainsi, le régulateur peut transmettre une information de puissance nulle ou maximale à l'unité de co-génération ;
- Chaudière : la chaudière est pilotée selon la demande de CO_2 de la serre et de l'état du stockage thermique. Lorsqu'il y a une demande de CO_2, la chaudière est démarrée à faible puissance puis stoppée si le stockage de chaleur est plein. À l'inverse, si le stockage thermique atteint un seuil bas, la chaudière est démarrée à pleine puissance. Le régulateur peut transmettre à la chaudière la puissance nulle, faible puissance ou puissance maximale ;
- Stockage thermique : une protection du stockage est présente sur ce régulateur. Lorsque le stockage est plein et que l'unité de co-génération produit de la chaleur, celle-ci est évacuée dans la serre agricole. Ainsi, il peut arriver de "surchauffer la serre agricole". Dans le cas d'une température trop importante dans la serre agricole (température de consigne plus $2°C$), les fenêtres de la serre sont ouvertes. Par conséquent, la chaleur excédentaire de l'unité de co-génération peu s'échapper librement dans l'atmosphère.

Les différentes variables à disposition pour le régulateur sont : la demande de la puissance de la chaudière et de l'unité de co-génération. Le régulateur peut transmettre une puissance nulle ou une puissance maximale pour l'unité de co-génération. Concernant la chaudière, ce régulateur peut transmettre une puissance nulle, une puissance minimale ou une puissance maximale. Les mesures constituant les états disponibles pour le régulateur sont le niveau de stockage de la chaleur par le stockage thermique et la demande de CO_2 de la serre. Par ailleurs,

le régulateur a aussi connaissance du mois de l'année. Une illustration du régulateur à base de règles expertes pilotant le processus est visible à la Fig.4.10, et le schéma de principe à la Fig.4.11.

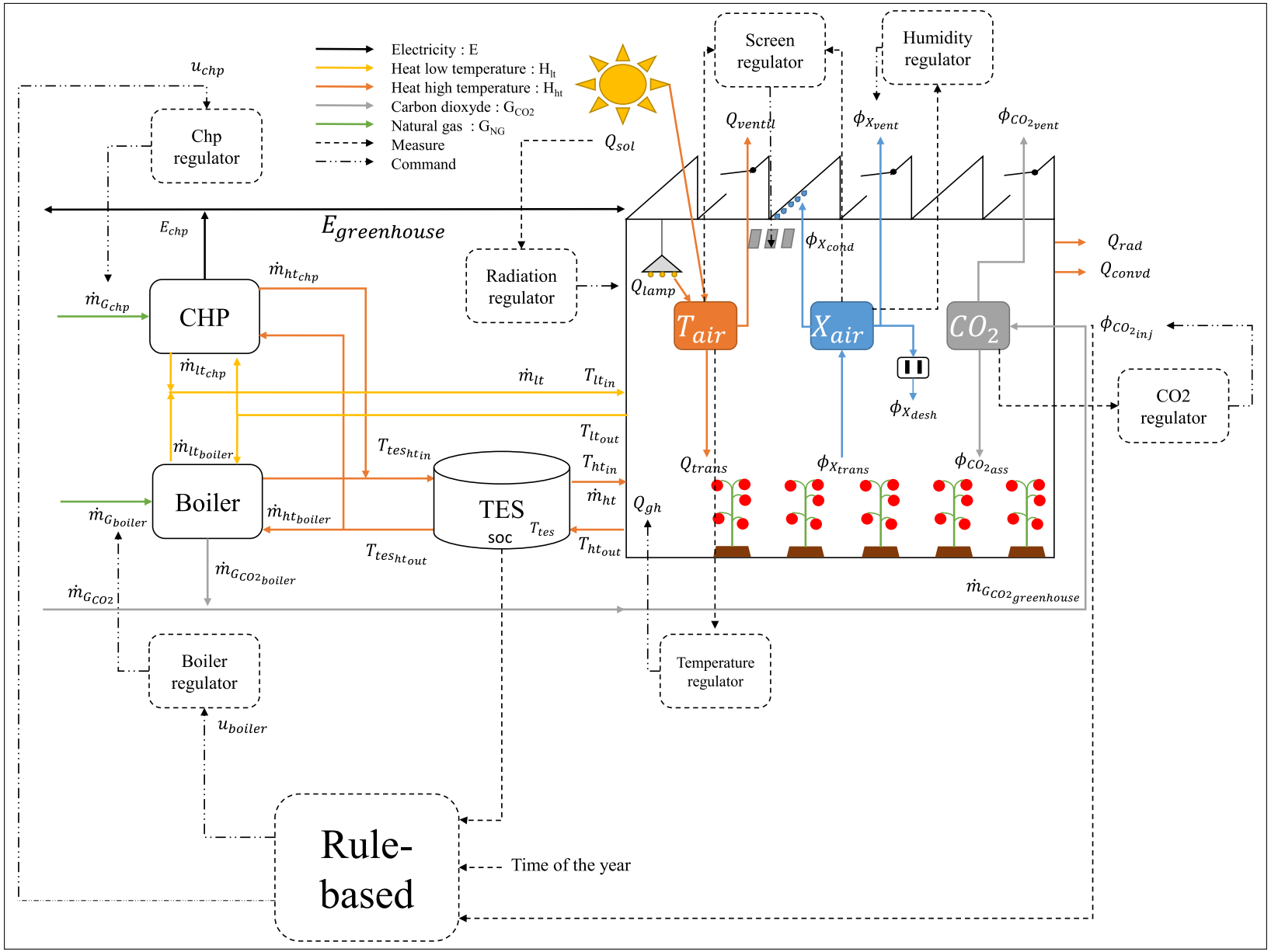

FIGURE 4.10 – Schéma de la serre agricole avec le régulateur classique.

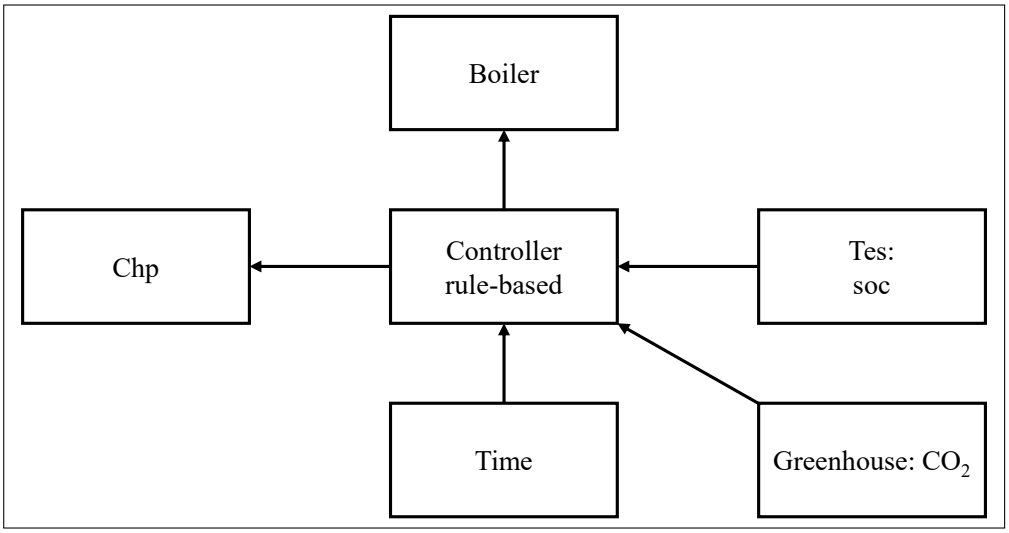

FIGURE 4.11 – Schéma de principe du régulateur à base de règles expertes et les variables à dispositions.

4.4.4 Pilotage avancé des équipements supports

Le pilotage avancé des équipements supports de la serre agricole a pour but d'améliorer la commande de ceux-ci et de proposer une alternative au pilotage classique. Les sections et paragraphes suivants présentent cette commande.

Commande hiérarchique La commande hiérarchique est basée sur deux niveaux de commande. Le premier niveau de commande réalise le pilotage des actionneurs du système à l'aide de la commande NN-MPC. Cette commande nécessite les consignes des équipements. Ce niveau reçoit les consignes du deuxième niveau de commande. Ce second niveau de commande réalise l'optimisation économique du système. Il calcule des points de consigne optimaux. Une illustration des deux niveaux de commande est visible à Fig.4.12. Les variables à disposition sont plus conséquentes que le régulateur à base de règles expertes. En effet, il est nécessaire de fournir les facteurs de conversion de la chaudière et de la co-génératrice, le bilan de puissance et les contraintes physiques et de fonctionnement. En outre, la demande énergétique de la serre agricole doit être prédite. Une illustration du régulateur des équipements de la serre avec la commande est visible à la Fig.4.13, puis le schéma de principe à Fig.4.14.

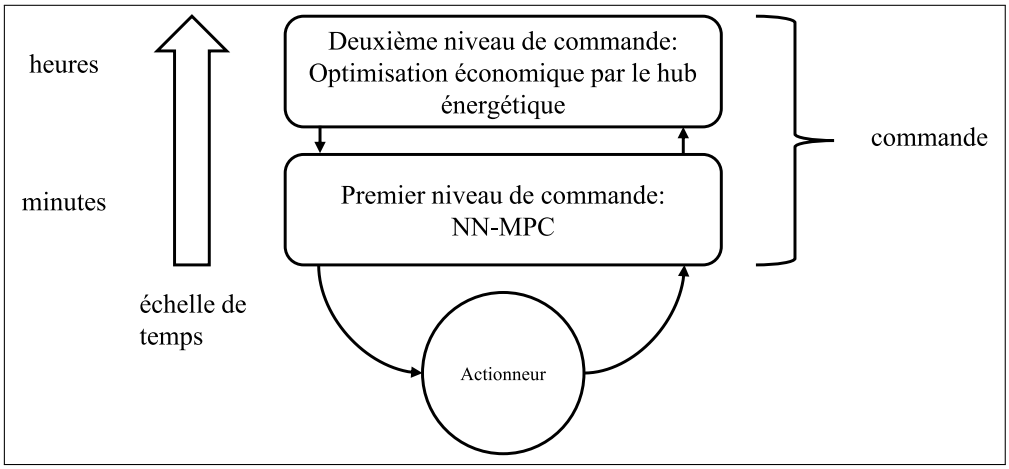

FIGURE 4.12 – Schéma de la commande avancée développée pour le système multi-énergies de la serre agricole.

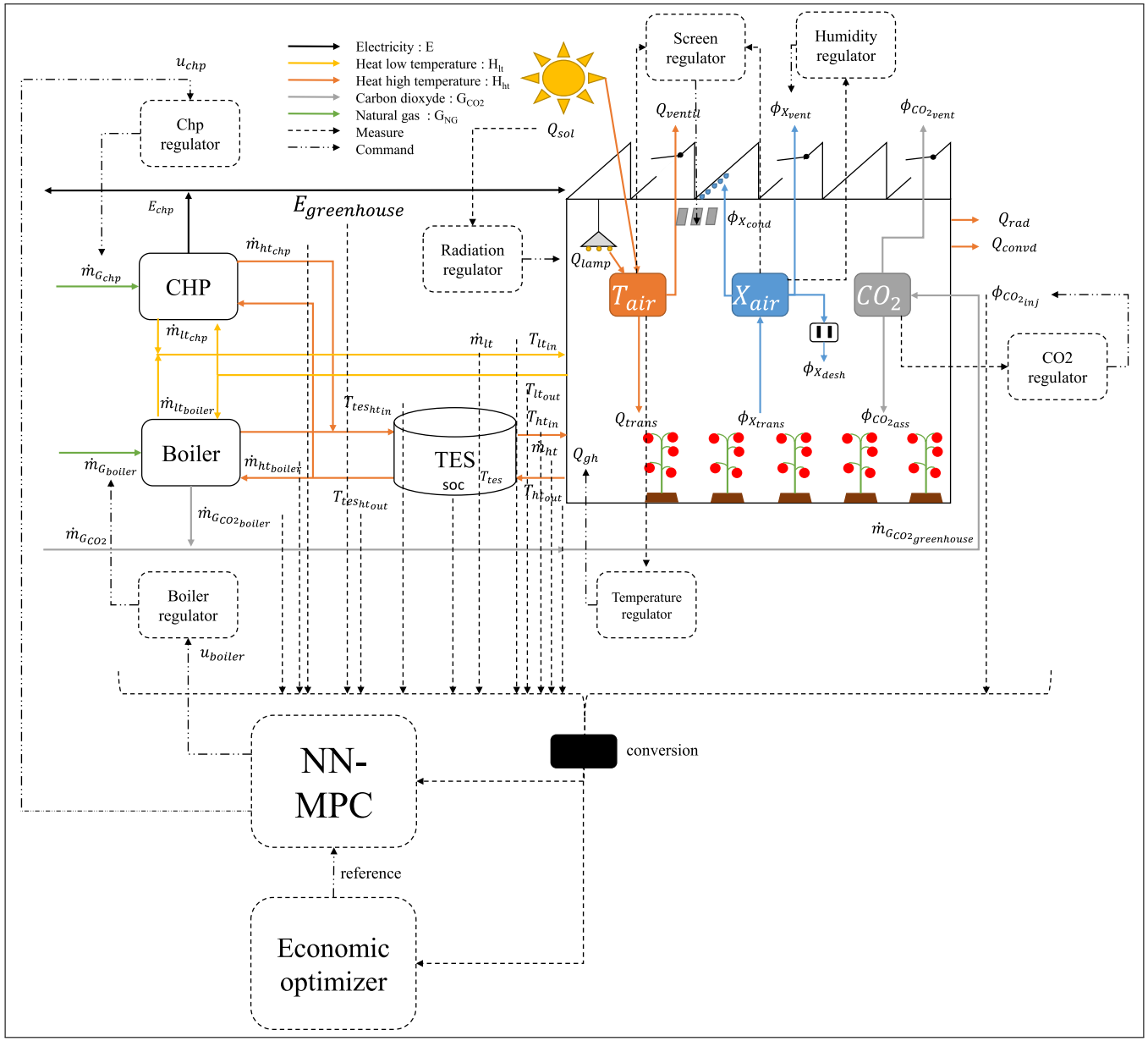

FIGURE 4.13 – Schéma de la serre agricole avec le régulateur avancé.

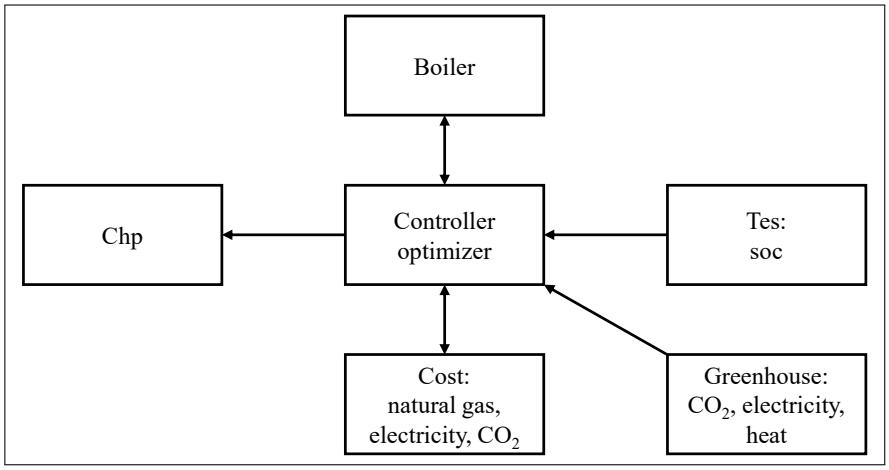

FIGURE 4.14 – Schéma de principe du régulateur avancé et les variables à dispositions.

Deuxième niveau de commande

Dans le cadre d'une projection post-contrat de rachat, l'électricité produite par l'unité de co-génération est valorisée soit par la consommation ou par la revente sur le marché. Pour répondre à cette problématique, le deuxième niveau de commande est développé. Cet étage réalise une optimisation économique avec la prise en compte des contraintes. En premier lieu, il est à noter qu'il n'y a pas de modélisation du système, ainsi les équations algébriques sont utilisées avec un bilan de puissance. Pour réaliser ce bilan, le cadre du hub énergétique est exploité. Le hub énergétique fut proposé par [79] pour fournir un cadre de modélisation de système multi-énergies, voir le chapitre 1 pour un rappel. En outre, les contraintes physiques et de fonctionnement associées aux équipements sont renseignées pour l'optimiseur. Une illustration des flux énergétiques avec une représentation par le hub énergétique est visible à la Fig.4.15.

Notation du hub énergétique Afin de réaliser le bilan de puissance du système multi-énergies et de faciliter la lecture, les notations suivantes sont utilisées [3] :

- I : $\overrightarrow{\text{vecteur}}$ des entrées des vecteurs énergétiques du hub énergétique ;
- L : $\overrightarrow{\text{vecteur}}$ des sorties des vecteurs énergétiques du hub énergétique. Il est associé aux demandes énergétiques et fertilisant de la serre agricole ;
- I_i : $\overrightarrow{\text{vecteur}}$ des entrées d'un équipement i de conversion ou de stockage (chaudière, co-génératrice, stockage) ;

3. $\overrightarrow{\text{vecteur}}$ est utilisé pour la notation mathématique puis vecteur pour les flux énergétiques.

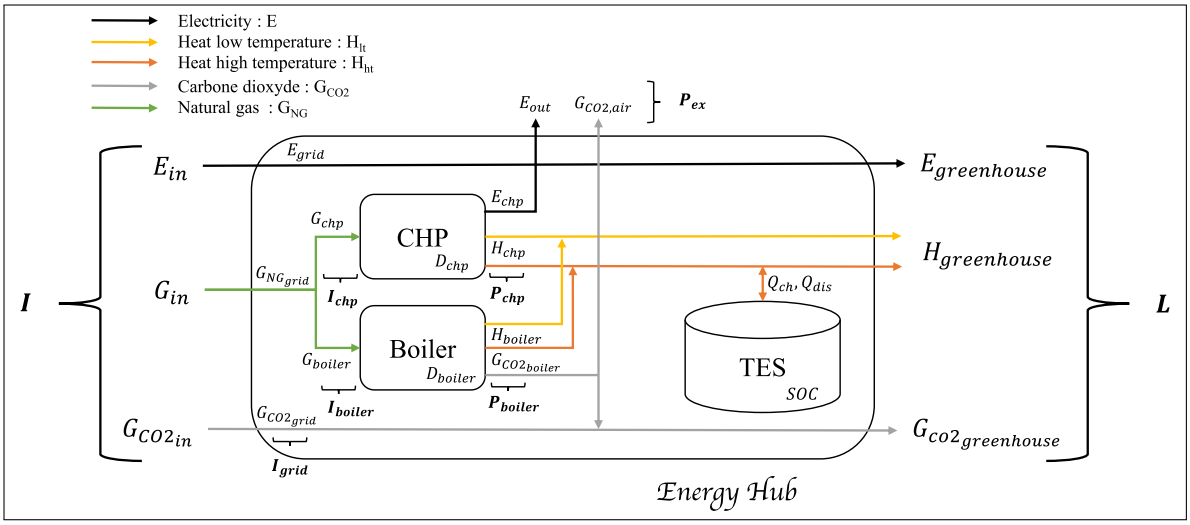

FIGURE 4.15 – Hub énergétique schématisé avec le système multi-énergies support de la serre agricole.

- P_i : $\overrightarrow{\text{vecteur}}$ des sorties d'un équipement de conversion i (chaudière, co-génératrice, stockage) ;
- $Cost_i$: variable spécifiant les coûts d'achat ou de vente énergétiques ;
- C : conversion d'un vecteur énergétique vers un second vecteur énergétique. Cette conversion est associée à un équipement. Dans notre cas d'étude, nous avons la chaudière et la co-génératrice ;
- D : matrice de conversion associée à un équipement (chaudière, co-génératrice, stockage thermique). Cette matrice contient les conversions C entre chaque vecteur. Pour simplification, les deux vecteurs de la chaleur sont joints pour former un unique vecteur. Les matrices D sont de dimension 3 colonnes et 3 lignes.

Nous pouvons maintenant ajouter les notations utilisées pour chaque vecteur énergétique et le vecteur fertilisant :

- E_i puissance électrique associée à un groupe énergétique i ;
- H_i puissance thermique associée à un groupe énergétique i ;
- G_i puissance de gaz naturel associée à un groupe énergétique i ;
- G_{CO2_i} débit de dioxyde de carbone associé à un groupe énergétique i ;

Bilan de puissance du hub énergétique Dans cette section le bilan de puissance du hub énergétique est formulé. Cette formulation est tirée de [232] avec une légère modification pour prendre en compte la vente de l'électricité et l'envoi du dioxyde de carbone en sortie de

la chaudière à l'atmosphère. Le vecteur de sortie L et le bilan de puissance associés au hub énergétique sont :

$$L = \begin{bmatrix} E_{greenhouse} \\ H_{greenhouse} \\ G_{CO2_{greenhouse}} \end{bmatrix} = P_{boiler} + P_{chp} + P_{grid} - P_{ex} - Q_{ch} + Q_{dis} \tag{4.6}$$

avec P_{boiler} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques en sortie de la chaudière, P_{grid} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques du réseau, P_{ex} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques expulsés du hub énergétique (la vente de l'électricité et le rejet de CO_2 dans l'air), Q_{ch} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques de chargement du stockage, Q_{dis} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques de déchargement du stockage, $E_{greenhouse}$ la puissance électrique de la demande de la serre, $H_{greenhouse}$ la puissance thermique de la demande de la serre, et $G_{CO2_{greenhouse}}$ le débit de dioxyde de carbone de la demande de la serre. Les $\overrightarrow{\text{vecteur}}$

présentés sont :

$$P_{boiler} = \begin{bmatrix} E_{boiler} \\ H_{boiler} \\ G_{CO2_{boiler}} \end{bmatrix} = D_{boiler} I_{boiler} \tag{4.7}$$

$$P_{chp} = \begin{bmatrix} E_{chp} \\ H_{chp} \\ G_{CO2_{chp}} \end{bmatrix} = D_{chp} I_{chp} \tag{4.8}$$

$$P_{grid} = \begin{bmatrix} E_{grid} \\ H_{grid} \\ G_{CO2_{grid}} \end{bmatrix} = D_{grid} I_{grid} \tag{4.9}$$

$$P_{ex} = \begin{bmatrix} E_{out} \\ 0 \\ G_{CO2_{air}} \end{bmatrix} \tag{4.10}$$

$$Q_{ch} = \begin{bmatrix} 0 \\ H_{tes,1} \\ 0 \end{bmatrix} \tag{4.11}$$

$$Q_{dis} = \begin{bmatrix} 0 \\ H_{tes,2} \\ 0 \end{bmatrix} \tag{4.12}$$

avec E_{boiler} la puissance électrique de la chaudière, celle-ci est fixée à zéro par la matrice D_{boiler}, H_{boiler} la puissance thermique de la chaudière, $G_{CO2_{boiler}}$ le débit de dioxyde de carbone de la chaudière. E_{chp} est la puissance électrique de la co-génératrice, H_{chp} est la puissance thermique produite et valorisée par la co-génératrice et $G_{CO2_{chp}}$ est le débit de CO_2 produit par l'unité de co-génération, ce débit est de zéro, car il n'y pas de valorisation de ce CO_2. E_{grid}, H_{grid} et $G_{CO2_{grid}}$ sont respectivement la puissance électrique, la puissance thermique en provenant d'un réseau de chaleur et le débit de CO_2 en provenance d'un réseau. H_{grid} est égal à zéro, car le hub énergétique n'est pas connecté à un réseau de chaleur. E_{out} est la puissance électrique réinjectée dans le réseau électrique et $G_{CO2_{air}}$ est le CO_2 produit par la chaudière qui n'est pas valorisé comme fertilisant et est expulsé dans l'air. $H_{tes,1_k}$ est la puissance de chargement du stockage de chaleur et $H_{tes,2_k}$ est la puissance de déchargement du stockage. Par ailleurs, D_{boiler}, D_{chp} et D_{grid} sont les matrices de conversion associées respectivement à la chaudière, à l'unité de

co-génération et au réseau, tel que :

$$D_{boiler} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & C_{boiler_{G_to_H}} & 0 \\ 0 & C_{boiler_{G_to_CO2}} & 0 \end{bmatrix} \tag{4.13}$$

$$D_{chp} = \begin{bmatrix} 0 & C_{chp_{G_to_E}} & 0 \\ 0 & C_{chp_{G_to_H}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.14}$$

$$D_{grid} = \begin{bmatrix} C_{grid_{to_E}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & C_{grid_{to_CO2}} \end{bmatrix} \tag{4.15}$$

avec $C_{boiler_{G_to_H}}$ la conversion du vecteur gaz naturel vers le vecteur chaleur par la chaudière, $C_{boiler_{G_to_CO2}}$ la conversion du vecteur gaz naturel vers le vecteur fertilisant de dioxyde de carbone par la chaudière, $C_{chp_{G_to_E}}$ la conversion du vecteur gaz naturel vers le vecteur électrique par l'unité de co-génération, $C_{chp_{G_to_H}}$ la conversion du vecteur gaz naturel vers la chaleur par l'unité de co-génération, $C_{grid_{to_E}}$ est la conversion du réseau électrique vers le vecteur électrique du hub énergétique et $C_{grid_{to_CO2}}$ la conversion du réseau de CO_2 vers le vecteur CO_2 du hub énergétique. $C_{grid_{to_E}}$ et $C_{grid_{to_CO2}}$ sont égaux à un, car il n'est pas pris en compte de perte éventuelle ou de transformateur de tension en entrée du hub énergétique. Par ailleurs, il est considéré pour simplification du cas d'étude que le vecteur CO_2 est connecté à un réseau avec une capacité infinie, et qu'il est en permanence capable d'assouvir le besoin. Alors que, les maraîchers possèdent des bouteilles de CO_2 et qu'il peut arriver qu'elles soient vides [4].

Le $\overrightarrow{\text{vecteur}}$ d'entrée I du hub énergétique associé aux vecteurs énergétiques d'entrée par le réseau et des entrées des équipements de conversion est :

$$I = \begin{bmatrix} E_{in} \\ G_{in} \\ G_{CO2_{in}} \end{bmatrix} = I_{grid} + I_{boiler} + I_{chp} \tag{4.16}$$

avec E_{in}, G_{in} et $CO2_{in}$ les vecteurs énergétiques en amont du hub énergétique, I_{grid}, I_{boiler} et I_{chp} les $\overrightarrow{\text{vecteur}}$ d'entrée associés au réseau, à la chaudière et la co-génératrice [5]. I_{grid}, I_{boiler} et I_{chp}

4. Une remarque équivalente est envisageable pour le réseau électrique et le réseau de gaz, ces éléments de considération sortent du périmètre de ce travail de recherche.

5. Il est nécessaire ici de créer un vecteur I_{grid} car le réseau peut fournir directement les charges de la serre en électricité ou en dioxyde de carbone. Ainsi du point de vue du hub énergétique ces flux proviennent du réseau.

sont égaux à :

$$I_{grid} = \begin{bmatrix} E_{grid} \\ G_{grid} \\ G_{CO2_{grid}} \end{bmatrix} \tag{4.17}$$

$$I_{chp} = \begin{bmatrix} 0 \\ G_{chp} \\ 0 \end{bmatrix} \tag{4.18}$$

$$I_{boiler} = \begin{bmatrix} 0 \\ G_{boiler} \\ 0 \end{bmatrix} \tag{4.19}$$

avec E_{grid} la puissance électrique ne provenant du réseau, G_{grid} la puissance de gaz naturel en provenance du réseau, $G_{CO2_{grid}}$ le débit de dioxyde de carbone en provenance du réseau, G_{chp} la puissance de gaz naturel utilisé par la co-génératrice et G_{boiler} la puissance de gaz naturel utilisé par la chaudière.

Ensuite, l'équation à la récurrence associée au stockage de la chaleur est renseignée :

$$SOC_{tes}[k+1] = SOC_{tes}[k] + \Delta_k H_{tes,1}[k] - \Delta_k H_{tes,2}[k] \tag{4.20}$$

avec SOC_{tes} l'état de charge du stockage thermique en MWh (*state of charge*), Δ_k le temps entre deux échantillons, $H_{tes,1}$ la puissance de chargement du stockage de chaleur et $H_{tes,2}$ la puissance de déchargement du stockage.

Contraintes de l'optimiseur

Plusieurs contraintes associées aux équipements (chaudière, co-génératrice, éléments de stockage) sont renseignées pour notre optimiseur. Ces contraintes sont extraites des spécificités des équipements énergétiques. Chaque contrainte est renseignée avec l'outil de modélisation des problèmes d'optimisation. Ces contraintes sont :

- Hub énergétique : le bilan de puissance du hub énergétique.
- Puissance minimale et maximale : les puissances minimale et maximale sont précisées pour la chaudière et la co-génératrice.

Par contre, du point de vue du réseau, I est une entrée du hub énergétique et la notation in est employée dans ce travail.

- Temps minimal de fonctionnement : le temps de fonctionnement minimal spécifie la durée minimale entre deux arrêts du groupe de production énergétique.
- Temps minimal d'arrêt : le temps minimal d'arrêt spécifie la durée minimale entre un arrêt et un démarrage du groupe de production énergétique.
- Rampe de puissance : la rampe maximale ou minimale de puissance spécifie la différence maximale ou minimale de puissance lors d'un changement de régime entre deux échantillons de temps.
- Stockage d'énergie : le stockage énergétique possède des contraintes physiques de capacité minimale et maximale pour emmagasiner de l'énergie.

Hub énergétique Le bilan de puissance du hub énergétique présenté à la section 4.4.4 est renseigné par des contraintes pour l'optimiseur.

Puissance minimale et maximale Les puissances des groupes énergétiques sont des variables semi-continues qui peuvent prendre une valeur entre un minimum et un maximum et zéro. Pour programmer ces variables dans le problème d'optimisation, des variables entières sont utilisées :

$$P_{min}\ State_k \leq P_k \leq P_{max}\ State_k \tag{4.21}$$

avec P_{min} un paramètre de la puissance minimale, P_{max} un paramètre de la puissance maximale, $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1) et k l'instant d'échantillonnage.

Temps minimal de fonctionnement Le temps minimal de fonctionnement est renseigné par la contrainte suivante :

$$\sum_{i=k-minUp+1}^{k} on_i \leq State_k \tag{4.22}$$

avec $minUp$ un paramètre indiquant le temps minimal de fonctionnement (multiple de l'échantillonnage), on_i est une variable continue permettant la détection des instants de démarrage du groupe énergétique, $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1) et k l'instant d'échantillonnage.

Temps minimal d'arrêt Le temps minimal d'arrêt est renseigné par la contrainte suivante :

$$\sum_{i=k-minDown+1}^{k} off_i \leq 1 - State_k \tag{4.23}$$

avec $minDown$ un paramètre indiquant le temps minimal d'arrêt (multiple de l'échantillonnage), off_i est une variable continue permettant la détection des instants d'arrêt du groupe énergétique, $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1) et k l'instant d'échantillonnage.

Rampe de puissance Lors d'un changement de régime du groupe énergétique, les contraintes permettent de limiter la rampe admissible. La contrainte de rampe maximale en puissance est :

$$\frac{P_{k+1} - P_k}{\Delta_k} \leq State_k \, RampUp \tag{4.24}$$

avec P la variable de puissance, $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1), Δ_k le temps entre deux échantillons, $RampUp$ un paramètre de la rampe maximale admissible. La contrainte de rampe minimale en puissance est :

$$\frac{P_{k+1} - P_k}{\Delta_k} \leq State_k \, RampDown \tag{4.25}$$

avec P la variable de puissance, $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1), Δ_k le temps entre deux échantillons, $RampDown$ un paramètre de la rampe minimale admissible.

Stockage d'énergie Les contraintes associées au stockage de la chaleur pour le cas d'étude sont :

$$SOC_{min} - SlackVariable \leq SOC_{tes} \leq SOC_{max} + SurplusVariable \tag{4.26}$$

avec SOC_{min} l'état de stockage minimal, SOC_{max} l'état de stockage maximal, SOC_{tes} la variable d'état de stockage, $SlackVariable$ et $SurplusVariable$ les variables de relâchement en contrainte molle. Ces variables permettent de ne pas stopper le problème d'optimisation dans le cas d'une violation de la contrainte.

Contraintes supplémentaires Des contraintes supplémentaires sont nécessaires lors de la modélisation du problème d'optimisation. Ces contraintes ne sont pas reliées aux limites physiques des équipements, mais concourent à la résolution. Tout d'abord, chaque variable du problème du bilan de puissance est positive :

$$0 \leq SlackVariable \tag{4.27}$$
$$0 \leq SurplusVariable \tag{4.28}$$
$$0 \leq P_{boiler} \tag{4.29}$$
$$0 \leq P_{chp} \tag{4.30}$$
$$0 \leq P_{grid} \tag{4.31}$$
$$0 \leq Q_{ch} \tag{4.32}$$
$$0 \leq Q_{dis} \tag{4.33}$$
$$0 \leq P_{ex} \tag{4.34}$$
$$0 \leq I \tag{4.35}$$
$$0 \leq I_{chp} \tag{4.36}$$
$$0 \leq I_{boiler} \tag{4.37}$$
$$0 \leq I_{grid} \tag{4.38}$$

Ensuite, une contrainte est ajoutée sur les variables On et Off pour les forcer à prendre des valeurs de 0 ou 1 :

$$State_k - State_{k-1} = On_k - Off_k \tag{4.39}$$

avec $State_k$ une variable binaire de l'état de fonctionnement à chaque instant (0 ou 1), on_i est une variable continue permettant la détection des instants de démarrage du groupe énergétique et off_i est une variable continue permettant la détection des instants d'arrêt du groupe énergétique.

Puis, la mesure de l'état de la batterie à l'instant de démarrage est renseignée avec une contrainte :

$$SOC_0 = SOC(t) \tag{4.40}$$

avec $SOC(t)$ est la mesure de l'état de stockage à l'instant d'exécution de l'optimisation et SOC_0 est l'état de stockage énergétique dans le problème d'optimisation.

Finalement, une contrainte terminale est ajoutée sur le stockage de la chaleur. Cette contrainte permet à l'optimiseur de ne pas décharger toute l'énergie stockée dans son horizon de calcul :

$$SOC_{f,1} - SlackVariable \leq SOC_N \leq SOC_{f,2} + SurplusVariable \tag{4.41}$$

avec $SOC_{f,1}$ et $SOC_{f,1}$ les contraintes terminales sur le stockage thermique avec des paramètres, $SlackVariable$ et $SurplusVariable$ les variables de relâchement en contrainte molle.

Fonction coût à minimiser

Le régulateur a pour rôle de minimiser les coûts économiques sur un horizon, avec des équations algébriques du bilan de puissance et des contraintes physiques et de fonctionnement des groupes énergétiques. Les différents coûts pris en compte dans cette étude sont :

- Coût de production : ce coût spécifie la dépense de fonctionnement d'un groupe énergétique.
- Coût de démarrage : ce coût spécifie la dépense de démarrage d'un groupe énergétique.

Coût de production Le coût de production est lié au tarif d'achat pour chaque vecteur énergétique ainsi que le gain apporté par la vente énergétique. Nous pouvons maintenant renseigner le coût du hub énergétique pour l'achat énergétique et de fertilisant :

$$Cost_1 = \begin{bmatrix} elec_{cost} \\ gas_{cost} \\ CO2_{cost} \end{bmatrix} I \tag{4.42}$$

avec $Cost_1$ les dépenses totales économiques du hub énergétique, $elec_{cost}$ le tarif de l'électricité, gas_{cost} le tarif du gaz naturel et $CO2_{cost}$ le tarif du dioxyde de carbone. Ensuite, le gain économique du hub énergétique pour la vente de l'électricité est :

$$Cost_2 = \begin{bmatrix} -elec_{cost} \\ 0 \\ 0 \end{bmatrix} P_{ex} \tag{4.43}$$

avec $Cost_2$ le gain de vente du hub énergétique, $elec_{cost}$ le tarif de l'électricité et P_{ex} $\overrightarrow{\text{vecteur}}$ contenant les vecteurs énergétiques de sortie du hub énergétique qui réalise une valorisation

énergétique.

Coût de démarrage La dépense économique résultant d'un démarrage d'un groupe énergétique est détectée avec la variable d'optimisation on :

$$Cost_{3_k} = on_k start_{cost} \tag{4.44}$$

avec $Cost_3$ le coût de démarrage d'un groupe énergétique, on_k est une variable continue permettant la détection des instants de démarrage du groupe énergétique, $start_{cost}$ le paramètre de dépense de démarrage d'un groupe énergétique.

Fonction de coût Il est nécessaire de conserver une fonction de coût convexe afin de garder le problème d'optimisation linéaire avec nombre entier (MILP). La fonction de coût contient le $\overrightarrow{\text{vecteur}}$ d'achat $Cost_1$, le $\overrightarrow{\text{vecteur}}$ de vente $Cost_2$ du hub énergétique et $Cost_{3_k}$ le coût de démarrage d'un équipement. Par ailleurs, il est nécessaire d'ajouter les variables $Slackvariable$ et $Surplusvariable$ pour la contrainte molle sur le stockage :

$$\begin{aligned} &\sum_{k=0}^{N-1} Cost_{1_k} + Cost_{2_k} + Cost_{3_k} \\ &\quad + \rho \sum_{k=0}^{N} Slackvariable_k + Surplusvariable_k \end{aligned} \tag{4.45}$$

avec $Cost_{1_k}$ le coût du hub énergétique, $Cost_2$ le gain de vente du hub énergétique, $Cost_{3_k}$ le coût de démarrage d'un groupe énergétique, $SlackVariable$ et $SurplusVariable$ les variables de relâchement en contrainte molle.

Premier niveau de commande

Commande MPC Le premier niveau de commande est réalisé à l'aide la commande MPC. Cette commande permet de piloter un système avec plusieurs entrées et sorties, la prise en compte des contraintes et la stabilité du système. De plus, la fonction considérée est une fonction de coût quadratique de minimisation des erreurs à la consigne. Les différents signaux à disposition pour le régulateur sont :

- Chaudière : puissance thermique totale produite et le débit de dioxyde de carbone produit ;
- Unité de co-génération : puissance thermique totale et électrique produite ;

— Stockage de la chaleur : le niveau de stockage de la chaleur ;

Ces différentes variables constituent les états pour la commande MPC. Ensuite, les actionneurs à disposition pour le régulateur sont :

— Chaudière : entrée de commande de la chaudière ;
— Unité de co-génération : entrée de commande de l'unité ;

En outre, des entrées de commande non-pilotage sont présentes avec le système multi-énergie. Ces entrées ont une action sur l'électricité produite et celle injectée au réseau, sur le stockage de la chaleur et sur le débit de CO_2 de la source. Ces entrées correspondent à la demande énergétique de la serre agricole :

— Serre agricole : Demande de chaleur, d'électricité et de CO_2 ;

La représentation du système dynamique en utilisant une équation récurrente discrète est :

$$\bar{x}[k+1] = f(\bar{x}[k], \bar{u}[k], \bar{v}[k]) \tag{4.46}$$

avec $\bar{x}$ le vecteur des états du système, $\bar{u}$ le vecteur des entrées de commande, $\bar{v}$ le vecteur des entrées non commandable et g le prédicteur des états du système. Ces vecteurs contiennent :

$$\bar{x}[k] = \begin{bmatrix} soc[k] \\ H_{boiler}[k] \\ H_{chp}[k] \\ E_{chp}[k] \\ G_{CO2_{boiler}}[k] \end{bmatrix}, \quad \bar{u}[k] = \begin{bmatrix} u_{boiler}[k] \\ u_{chp}[k] \end{bmatrix}, \quad \bar{v}[k] = \begin{bmatrix} H_{greenhouse}[k] \\ E_{greenhouse}[k] \\ G_{CO2_{greenhouse}}[k] \end{bmatrix} \tag{4.47}$$

avec soc le niveau de stockage thermique, H_{boiler} la puissance thermique de la chaudière, H_{chp} la puissance thermique de l'unité de co-génération, E_{chp} la puissance électrique de l'unité de co-génération, $G_{CO2_{boiler}}$ le débit de CO_2 de la chaudière, u_{boiler} l'entrée de commande de la chaudière, u_{chp} l'entrée de commande de l'unité de co-génération, $H_{greenhouse}$ l'entrée non pilotable de la puissance thermique de la serre agricole, $E_{greenhouse}$ l'entrée non pilotable de la puissance électrique de la serre agricole et $G_{CO2_{greenhouse}}$ l'entrée non pilotable de la consommation de CO_2.

La méthodologie développée dans la section 2.1 est utilisée pour synthétiser le régulateur du premier niveau de commande hiérarchique. Les différentes étapes de la méthodologie sont présentées dans les sections suivantes. Ces étapes sont : génération et récolte des données, paramétrisation du modèle boîte noire et synthèse du régulateur.

Génération et récoltes des données Les données sont nécessaires pour entraîner le réseau de neurones afin de prédire les états du système. Durant ce travail, les données utilisées proviennent des deux simulations. La première provient de la simulation du pilotage du système numérique avec les règles expertes présentées dans la section. Une seconde simulation est réalisée avec le premier niveau de la commande hiérarchique par l'optimisation des coûts économiques. Durant ces deux simulations, les données ont été sauvegardées avec un pas de temps de 1s. Nous pouvons noter ces données par D. Les données sont filtrées pour obtenir cinq groupes discrétisés selon le temps d'échantillonnage. En effet, le temps d'échantillonnage est modifié pour obtenir des données à 10s, 30s, 60s, 120s et 300s : D10s, D30s, D60s, D120s, D300s. Pour chaque jeu de données, il y a la séparation entre les données pour l'entraînement puis pour le test et vérification. Les données de validation sont extraites aux données d'entraînements tout les 100 epochs.

Paramétrisation du modèle boîte noire Le modèle boîte noire est employé pour réaliser les prédictions des états du système au prochain instant d'échantillonnage $k+1$, tel que :

$$\hat{x}[k+1] = g_{nn}(\bar{x}[k], \bar{u}[k], \bar{v}[k]) \tag{4.48}$$

avec, $\hat{x}[k+1]$ les états prédits de $\bar{x}[k+1]$, $\bar{u}$ les entrées de commande $\bar{v}$ les entrées exogènes et g_{nn} le modèle neuronale. Dans ce travail, le réseau ResNet est considéré avec une couche et il est constitué d'un sous réseau Fnn, de plus les couches d'entrée et de sortie utilisent des neurones avec une fonction d'activation identité. Une illustration du réseau est visible à la Fig.4.16.

Synthèse du régulateur La commande prédictive utilisée pour piloter le système serre est une MPC contractive. Elle possède une fonction coût quadratique et une contraction des états finaux pour assurer la stabilité et faisabilité [152] :

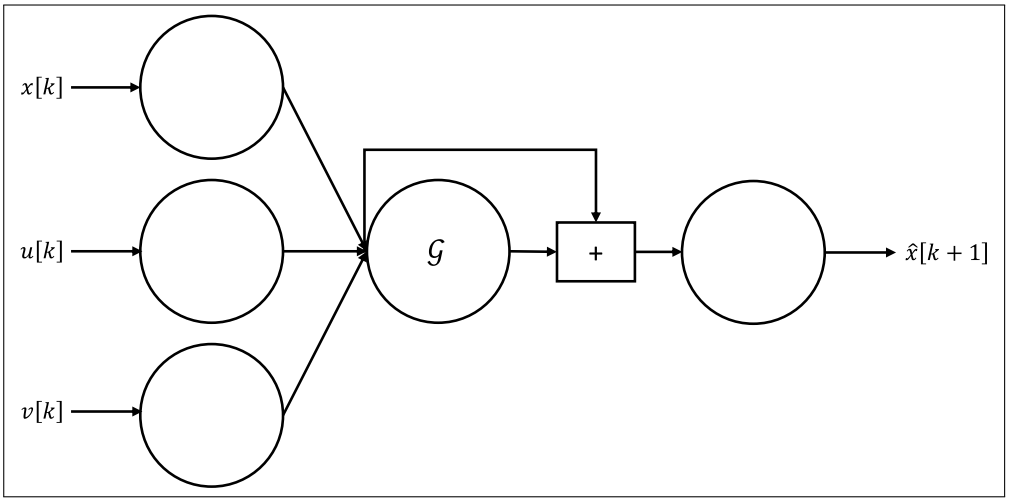

FIGURE 4.16 – Réseau de neurones utilisé pour la prédiction des états du système multi-énergie.

$$\min_{\tilde{u},\tilde{x}} \sum_{i=0}^{N-1} \tilde{x}_i^T Q \tilde{x}_i + \tilde{u}_i^T R \tilde{u}_i \tag{4.49}$$

$$\text{s.c. } \hat{x}_{i+1} = g_{nn}^*(\hat{x}_i, u_i, \bar{v}_i) \tag{4.50}$$

$$\hat{x}_i \in \mathcal{X} \tag{4.51}$$

$$u_i \in \mathcal{U} \tag{4.52}$$

$$\hat{x}_0 = \bar{x}(t) \tag{4.53}$$

$$\tilde{x}_N P \tilde{x}_N \leq \alpha \tilde{x}_0 P \tilde{x}_0 \tag{4.54}$$

$$\tilde{x}_i = \hat{x}_i - x_i^r \tag{4.55}$$

$$\tilde{u}_i = u_i - u_i^r \tag{4.56}$$

avec l' Eq.4.49 la fonction coût quadratique à minimiser. $\tilde{u}$ est la séquence d'entrée aux écarts : $\tilde{u} := \tilde{u}_0, \ldots, \tilde{u}_{N-1}$ et $\tilde{x}$ est la séquence d'état aux écarts : $\tilde{x} := \tilde{x}_0, \ldots, \tilde{x}_N$ avec N l'horizon. L'Eq.4.50 est l'équation d'évolution des états, elle est tirée du prédicteur à base de réseau de neurones artificiels. Les contraintes sur les états et les entrées de commande sont visibles aux Eq.4.51 et Eq.4.52. Eq.4.53 concerne l'initialisation des états avec la mesure des états du système avant l'exécution du problème d'optimisation. Eq.4.54 présente les contraintes finales avec les contractions sur les états. Finalement, l'Eq.4.55 et l'Eq.4.56 sont les erreurs des états et des entrées de commande par rapport aux consignes.

4.4.5 Plan d'expérience

Simulateur

La modélisation fine du système serre agricole est exportée en *Functional Mock-up Unit* (FMU) en utilisant le protocole *Functional Mock-up Interface* (FMI) [201]. La FMU est simulée dans un environnement de simulation pour réaliser les connexions avec le régulateur des règles expertes programmées. L'environnement de simulation sélectionné est Simulink de Matlab. Par ailleurs, les données météorologiques utilisées pour l'année de simulation est 2019, elles sont extraites de la base de données : *Modern-Era Retrospective analysis for Research and Applications, Version 2 (MERRA-2)* [233]. Les variables utilisées sont : la température extérieure, la température du ciel, l'humidité relative, le rayonnement solaire, la vitesse du vent et la concentration de dioxyde de carbone de l'air, Fig.4.17.

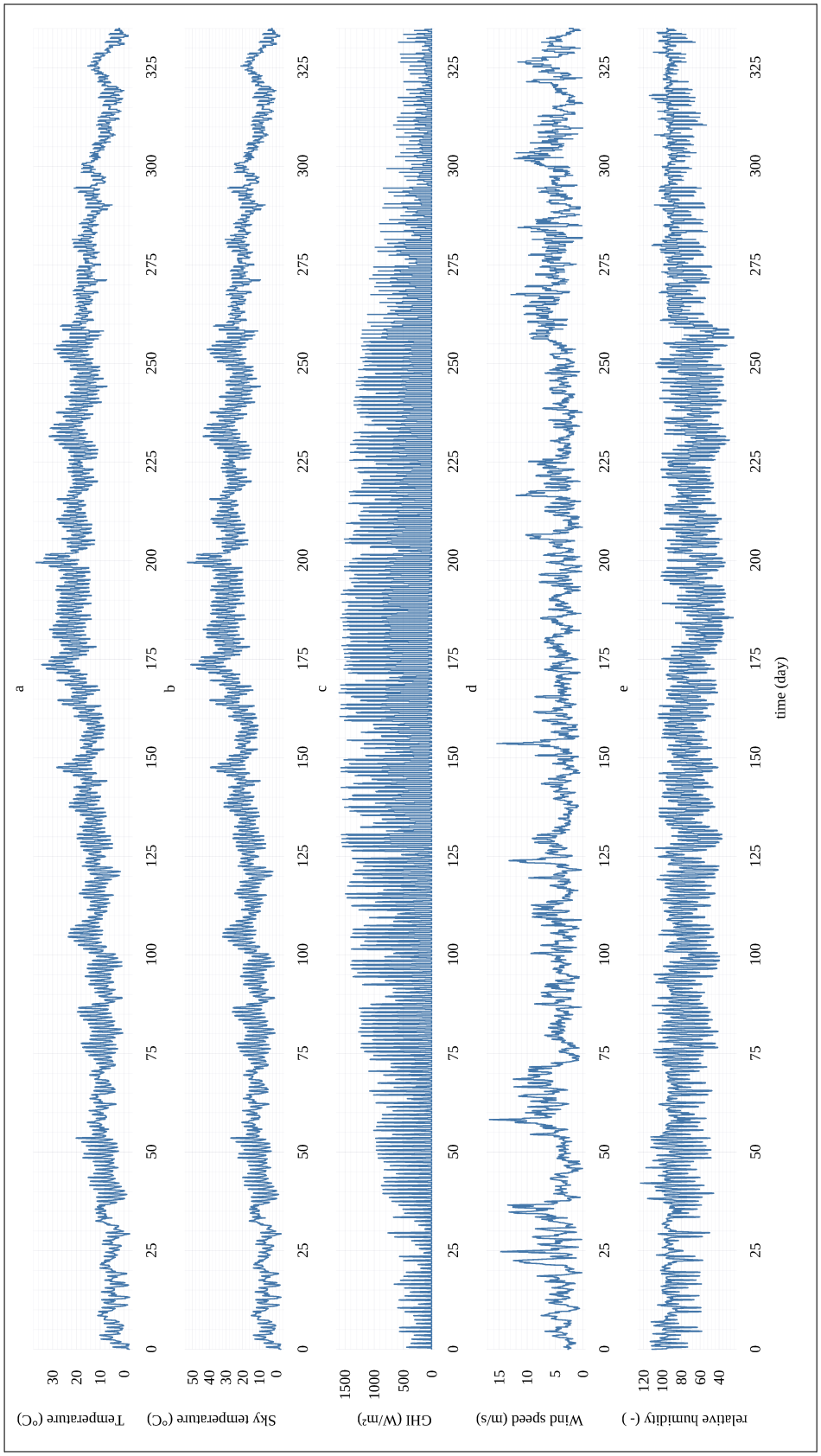

FIGURE 4.17 – Variables météorologiques de l'année 2019 avec (a) la température extérieure, (b) la température du ciel, (c) le rayonnement global, (d) la vitesse du vent et (e) l'humidité relative.

Une illustration de l'environnement de simulation avec la FMU est visible à la Fig.4.18.

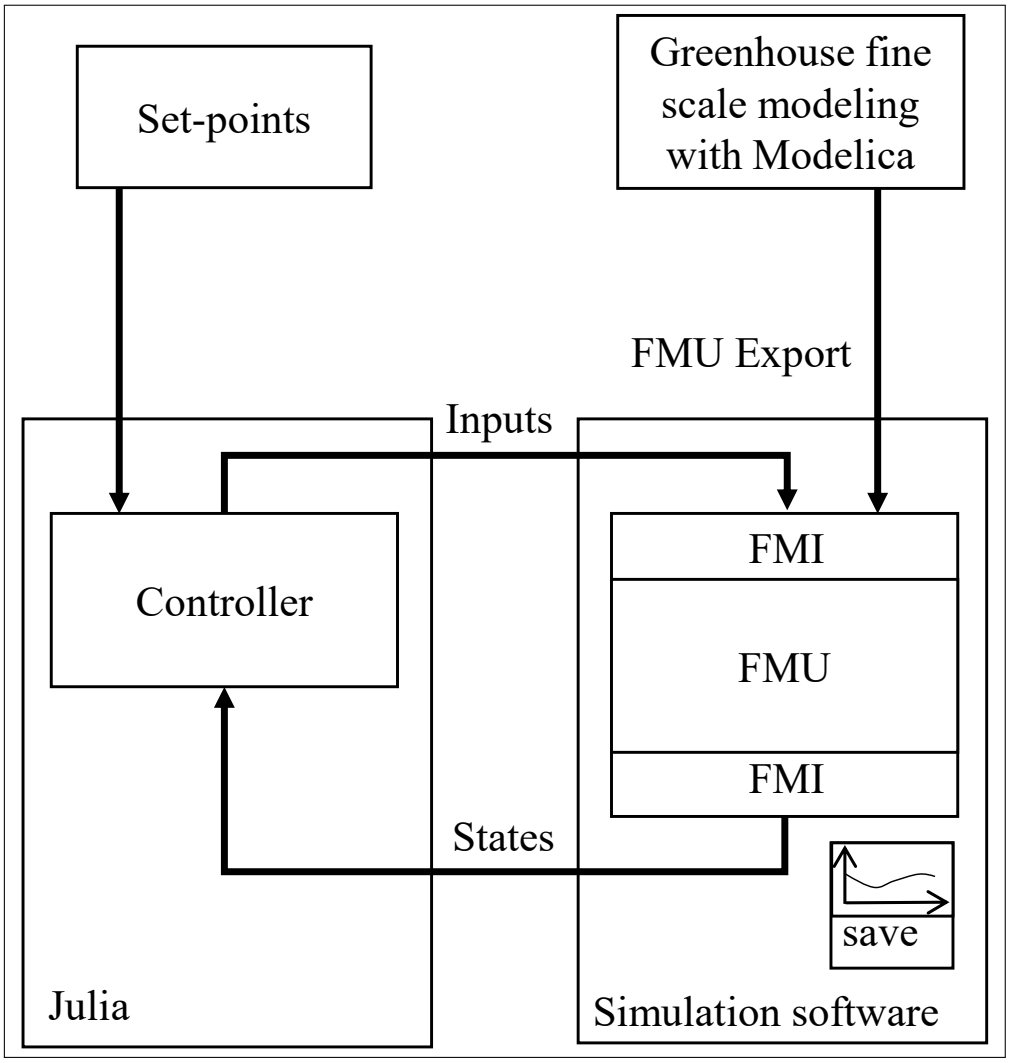

FIGURE 4.18 – Environnement numérique de simulation contenant la FMU et un régulateur.

Pilotage classique

Le régulateur est implémenté avec des portes logiques et des hystérésis dans simulink. L'information du mois de l'année est transmise par le simulateur, puis le régulateur transmet la mise en marche ou l'arrêt de la chaudière et de l'unité de co-génération.

Pilotage avancé

Deuxième niveau de commande Le deuxième niveau de commande est formulé avec l'outil JuMP [164] avec le langage Julia [165]. Les contraintes sur l'unité de co-génération engendrent des variables entières et par conséquent le problème d'optimisation est *Mixed Integer Linear Programming (MILP)*. Il existe plusieurs solveurs à disposition pour le résoudre. Nous pouvons citer *Coin-or branch and cut* (Cbc), utilisant le solveur linéaire *Coin-or linear programming* (CLP) de l'initiative open source Coin-or [234] avec la méthode *branch and cut* [235]. Le solveur utilisé dans ce travail est *IBM CPLEX optimizer*. CPLEX permet pour un résultat équivalent à Cbc de réduire le temps de résolution. Cette réduction est obtenue par des méthodes de simplification automatique du problème d'optimisation : réduction du nombre de variables entières, normalisation, heuristique de réduction du problème d'optimisation avec l'algorithme *branch and cut* et une parallélisation sur plusieurs cœurs du microprocesseur de l'algorithme de résolution. L'optimiseur possède un horizon de 24 heures et un pas de temps de 15 minutes. L'optimisation est exécutée toutes les heures. Pour réaliser la prédiction des demandes, une simulation a été accomplie avec les données météorologiques de 2019, ainsi les demandes énergétiques de la serre agricole ont été enregistrées. Par conséquent, pour le cas de la simulation avec le régulateur optimiseur, les prévisions des demandes énergétiques sont parfaites. Les paramètres renseignés pour l'optimisation sont présentés avec trois tableaux. Premièrement, les paramètres des contraintes sont visibles à la Tab.4.2. Deuxièmement, les coûts économiques renseignés sont visible à la Tab.4.3. Troisièmement, les paramètres des matrices de conversion des équipements énergétiques sont visibles à la Tab.4.4.

Premier niveau de commande Le modèle boîte noire est paramétrisé par l'algorithme de rétropropagation du gradient et les hyperparamètres sont optimisés avec un algorithme metaheuristique. L'implémentation informatique utilise le langage Julia et des boîtes à outils dédié à l'apprentissage machine et à l'optimisation [236]. La boîte à outil utilisée pour le modèle boîte noire est Flux [136] et l'optimisation des hyperparamètres à l'aide de la librairie BlackBoxOptim [133]. La fonction coût sélectionnée est l'erreur quadratique moyenne, voir section 2.4.4. Les hyperparamètres autorisés pour l'algorithme metaheuristique sont visibles à la Tab 4.5. L'algorithme est arrêté dès que 50 réseaux sont entrainés pour chacun des jeux de paramètres soit un total 250. À chaque itération l'algorithme a la possibilité de tester puis évaluer les hyperparamètres occasionnant une réduction de l'erreur quadratique moyenne. Par ailleurs, l'entraînement des réseaux est réalisé sur GPU avec une machine de calcul. En outre, après l'application de l'algorithme, un traitement est réalisé pour enlever les réseaux dont la

TABLE 4.2 – Paramètres renseignés pour le problème d'optimisation.

Paramètres	Chaudière	Co-génération	Stockage chaleur
Puissance d'arrêt	0	0	-
Puissance minimale de fonctionnement	0.5×10^6	2.2×10^6	-
Puissance maximale de fonctionnement	6.5×10^6	4.4×10^6	-
Rampe minimale de démarrage	-	-	-
Rampe maximale de démarrage	-	-	-
Temps de fonctionnement minimal	1 heure	5 heures	-
Temps d'arrêt minimal	1 heure	10 heures	-
Puissance maximale de charge (puissance maximale de la chaudière et de la co-génération multiplié par 0.9)	-	-	$9.9MWh$
Puissance maximale de décharge	-	-	inf
Stockage minimal	-	-	102 MWh
Stockage maximal	-	-	150.8 MWh
Contrainte terminale minimale	-	-	121 MWh
Contrainte terminale maximale	-	-	131 MWh

TABLE 4.3 – Paramètres des coûts économiques renseignés pour le problème d'optimisation.

Paramètres	Chaudière	Co-génération	Réseau
Gaz naturel	-	-	20 €/MW
Coût fixe de démarrage	-	200 €	-
Coût de dioxyde de carbone	-	-	150 €/T
Coût et vente électrique	-	-	Variant dans le temps, Fig.4.19 et Fig.4.20.

TABLE 4.4 – Paramètres des facteurs de conversion des équipements.

Paramètres	Valeurs	Unités
$C_{boiler_{G_to_H}}$	0.92	(-)
$C_{boiler_{G_to_CO2}}$	5.7e-8	(-)
$C_{chp_{G_to_E}}$	0.35	(-)
$C_{chp_{G_to_H}}$	0.35	(-)
$C_{grid_{to_E}}$	1	(-)
$C_{grid_{to_CO2}}$	1	(-)

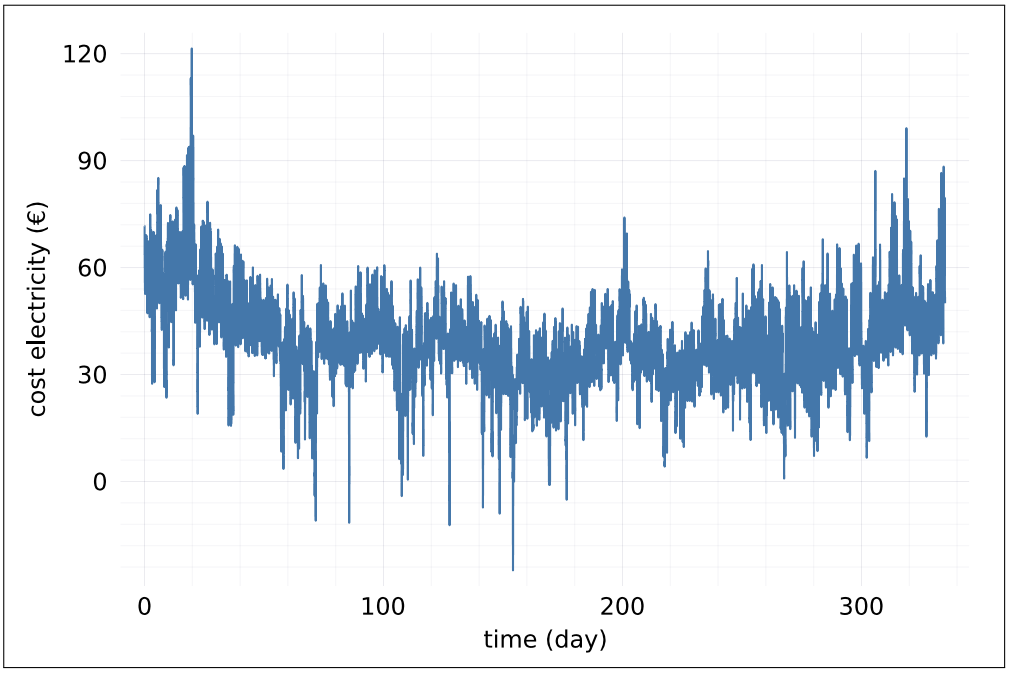

FIGURE 4.19 – Coût de l'électricité un jour à l'avance sur l'année 2019.

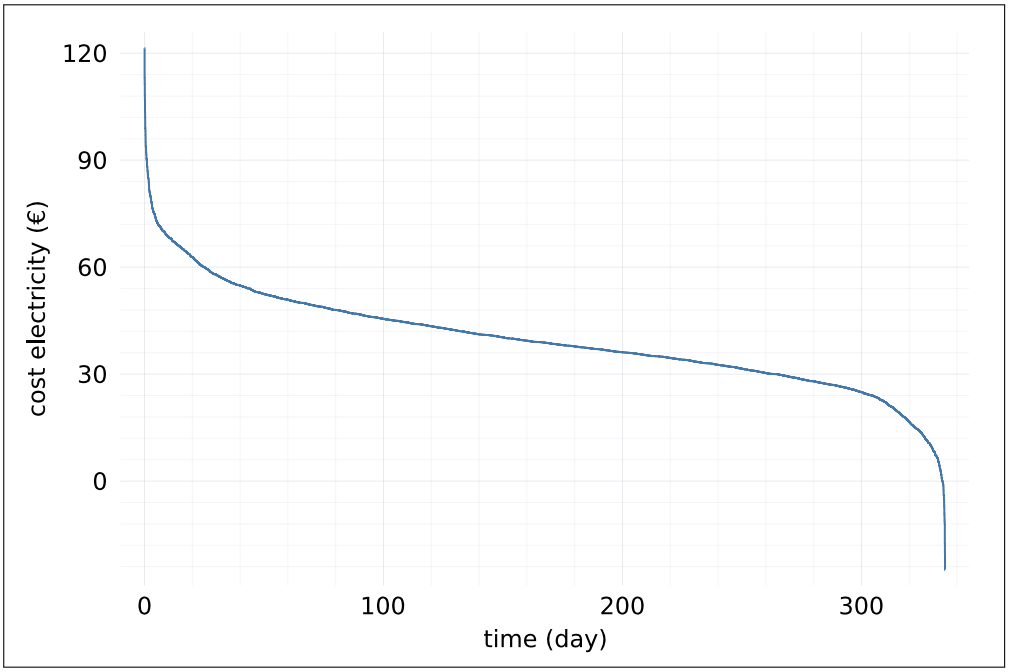

FIGURE 4.20 – Monotone des coûts électriques sur l'année 2019.

TABLE 4.5 – Hyperparamètres autorisés du réseau de neurones.

Paramètres	Valeurs autorisées
Fonctions d'activations	Relu [195]
Neurones	5 à 15
Couches cachées	1 à 3
Couche d'intégration pour ResNet	1
Époques	100 à 500
Batch size	1024
Optimiseur	Adam [197], Radam [198], Nadam [199], Oadam [200]
Taux apprentissage	1×10^{-7} à 0.001
Momentum décroissance exponentiel	0.9 à 0.999
Momentum estimation	0.9 à 0.999

disparition du gradient est arrivée (fonction coût MSE plus grande que 1).

Le premier niveau de commande est implémenté avec l'outil JuMP [164] dans le langage Julia [165]. L'outil JuMP permet de renseigner une fonction informatique Julia pour calculer automatiquement le gradient, cette méthode est intitulée *User-defined Functions*. Dans notre cas d'étude, la fonction renseignée dans l'outil est le prédicteur neuronal. Le problème d'optimisation est non-linéaire et un solveur adéquate est nécessaire. Le solveur de résolution utilisé est Knitro [172]. Par ailleurs, nous avons sélectionné le prédicteur réalisant la plus faible MSE avec la période d'échantillonnage de $10s$. En outre, la MPC est exécutée tous les $300s$ avec un pas entre deux échantillons de $300s$. Pour utiliser le prédicteur échantillonné à $10s$, une boucle *for* de 30 pas est réalisée avec l'*User-defined Functions* de l'outil JuMP. Par ailleurs, les paramètres de la MPC sont visibles à la Tab.4.6.

4.5 Résultats

4.5.1 Apprentissages des réseaux de neurones

Fig.4.21 expose les résultats de l'apprentissage des réseaux avec les boîtes à moustaches présentant la fonction coût en fonction des pas d'échantillonnage de $10s$, $30s$, $60s$, $120s$ et $300s$ avec les données de validation (Fig.4.21.a) et de test (Fig.4.21.b). Il est observé que l'augmentation du temps d'échantillonnage occasionne une augmentation de la fonction de coût. La plus faible MSE est observée pour $10s$ avec la fonction de coût médiane de 9.40×10^{-7} alors que la médiane est de 8.22×10^{-6} pour $30s$, 5.49×10^{-5} pour $60s$, 2.63×10^{-4} pour $120s$ et 1.54×10^{-3} pour $300s$

TABLE 4.6 – Paramètres de la MPC.

Paramètres	Valeurs
N	12
ΔT	$300s$
Q	$\begin{bmatrix} 5 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
R	$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$
P	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 & 0 \\ 0 & 0 & 0.001 & 0 & 0 \\ 0 & 0 & 0 & 0.001 & 0 \\ 0 & 0 & 0 & 0 & 0.001 \end{bmatrix}$
α	1
Temps optimisation maximal	$30s$
Solveur de résolution	Knitro [172]
Contraintes sur les états	[0 1]
Contraintes sur les entrées de commande	[0 1]

avec les données de validation. La synthèse des médianes est visible aux Tab.4.7 et Tab.4.8 pour les données de validation et de test.

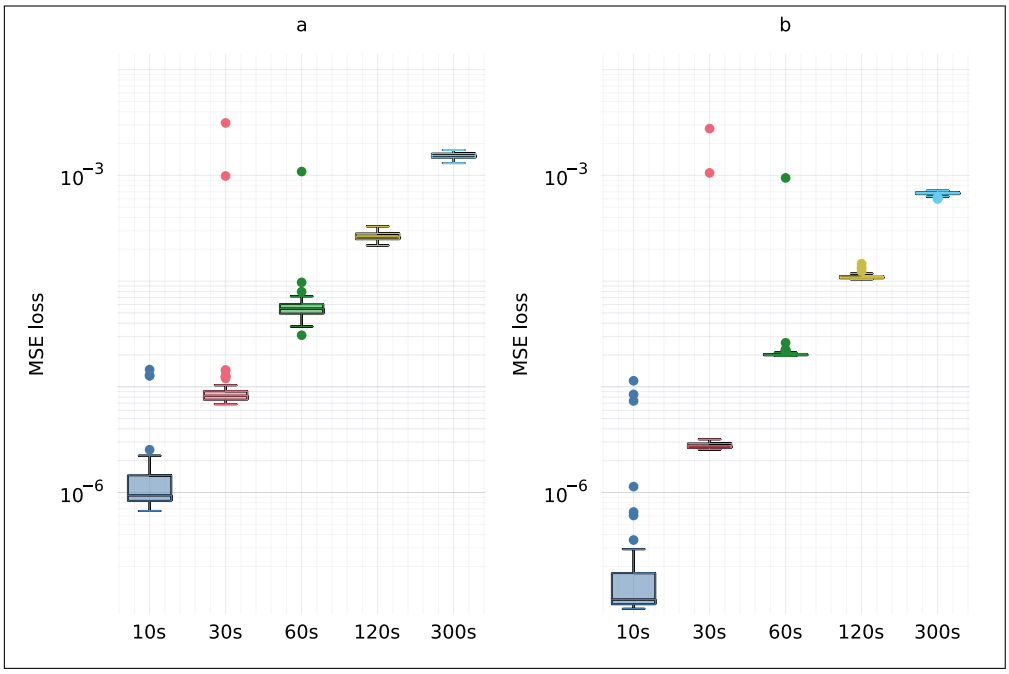

FIGURE 4.21 – boîte à moustache de la fonction de coût selon les temps d'échantillonnage des données. (a) Présente les boîtes à moustache avec les données de validation. (b) Présente les boîtes à moustache avec les données de test.

TABLE 4.7 – Fonction de coût avec les données de validation.

Temps d'échantillonnage	Mediane	Premier quartile	Quatrième quartile
$10s$	9.40×10^{-7}	8.34×10^{-7}	1.45×10^{-6}
$30s$	8.22×10^{-6}	7.55×10^{-6}	9.09×10^{-6}
$60s$	5.49×10^{-5}	4.92×10^{-5}	6.07×10^{-5}
$120s$	2.63×10^{-4}	2.49×10^{-4}	2.83×10^{-4}
$300s$	1.54×10^{-3}	1.47×10^{-3}	1.61×10^{-3}

Fig.4.22 présente les résultats de l'apprentissage des réseaux avec les boîtes à moustaches avec la fonction de coût selon les hyperparamètres pour les données de validation échantillonnées à $10s$. Fig.4.22.a présente la fonction de coût en fonction des couches cachées, Fig.4.22.b présente la fonction de coût en fonction du nombre de neurones par couche cachée, Fig.4.22.c présente la fonction de coût en fonction du nombre d'epochs et Fig.4.22.d présente la fonction de coût en fonction de l'optimiseur. Concernant les Fig.4.22.a et Fig.4.22.d il n'est pas observé d'influence significative pour la réduction de la MSE avec l'augmentation du nombre de couches

TABLE 4.8 – Fonction de coût avec les données de test.

Temps d'échantillonnage	Median	Premier quartile	Quatrième quartile
$10s$	9.71×10^{-8}	8.76×10^{-8}	1.73×10^{-7}
$30s$	2.76×10^{-6}	2.64×10^{-6}	2.91×10^{-6}
$60s$	2.01×10^{-5}	1.99×10^{-5}	2.05×10^{-5}
$120s$	1.09×10^{-4}	1.06×10^{-4}	1.12×10^{-4}
$300s$	6.79×10^{-4}	6.65×10^{-4}	6.93×10^{-4}

ou la sélection d'un optimiseur adam, radam, oadam ou radam. Tandis que la sélection du nombre de neurones à une influence sur la fonction de coût. Il est observé avec la Fig.4.22.b que l'augmentation du nombre de neurones jusqu'à 8 par couche caché permet la réduction de la fonction de coût médiane. De plus, concernant le nombre d'epochs, il est remarqué que l'augmentation au-delà de 200 permet la réduction de la fonction de coût, Fig.4.22.c. En effet, il est remarqué un plateau de la fonction de coût avec 300, 400 et 500 epochs.

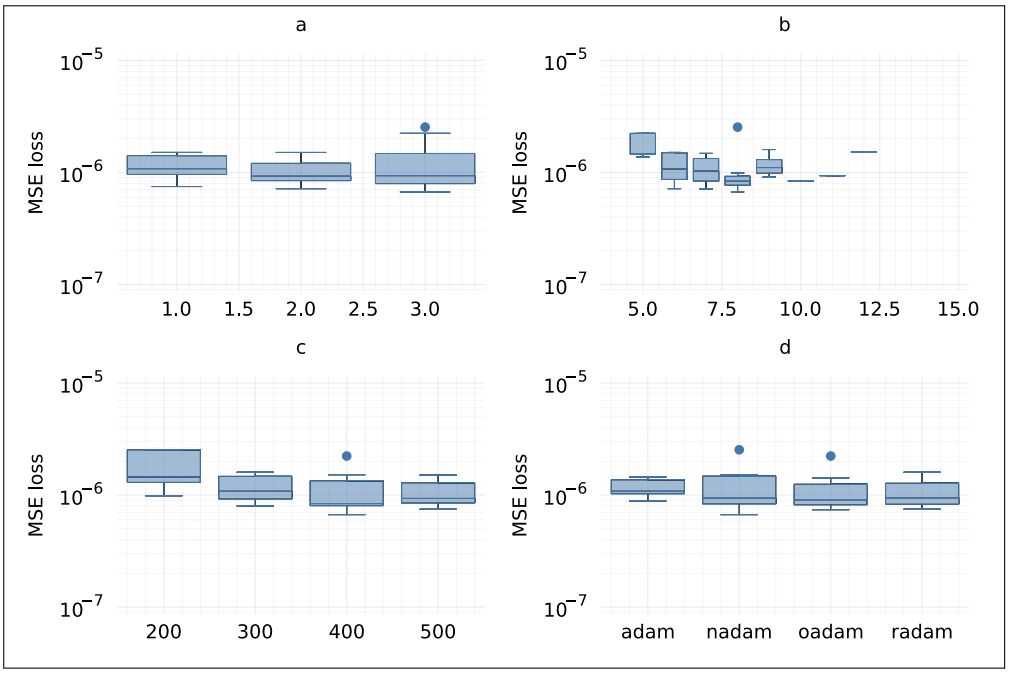

FIGURE 4.22 – boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $10s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur.

Fig.4.23 présente les résultats de l'apprentissage des réseaux avec les boîtes à moustaches avec la fonction de coût selon les hyperparamètres pour les données de validation échantillon-

nées à 30s. Fig.4.23.a présente la fonction de coût en fonction des couches cachées, Fig.4.23.b présente la fonction de coût en fonction du nombre de neurones par couche cachée, Fig.4.23.c présente la fonction de coût en fonction du nombre d'epochs et Fig.4.23.d présente la fonction de coût en fonction de l'optimiseur. Il est observé un plateau de la fonction selon les hyperparamètres, nombre de couches cachées Fig.4.23.a, nombre de neurones par couches cachées Fig.4.23.b, nombre d'epochs Fig.4.23.c et l'optimiseur Fig.4.23.d.

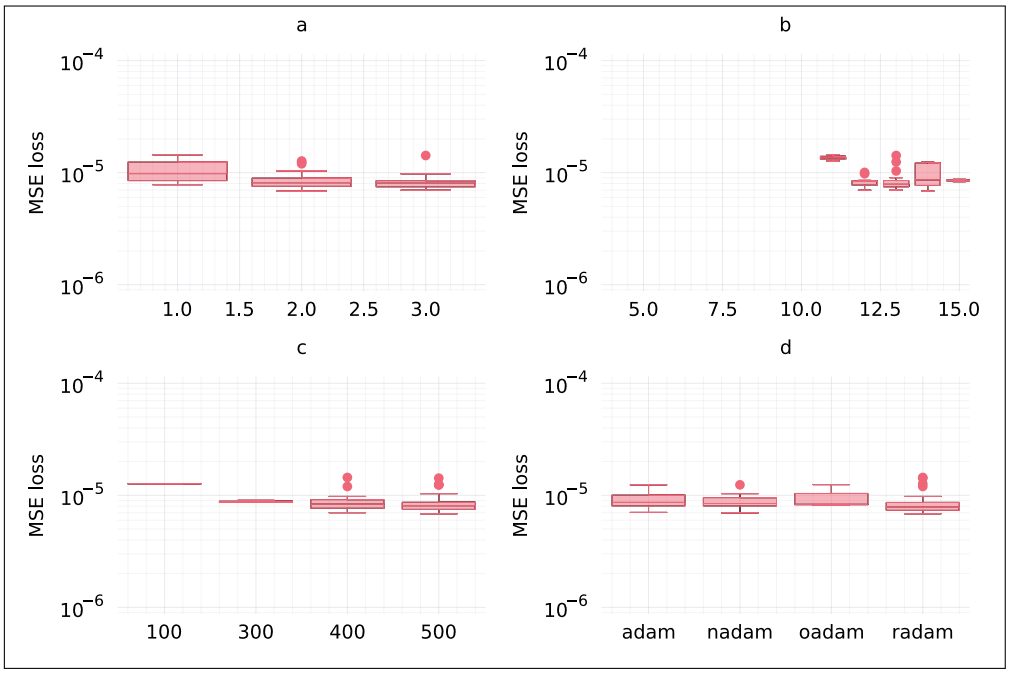

FIGURE 4.23 – boîte à moustache de la fonction de coût avec les données de validation échantillonnées à 30s selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur.

Fig.4.24 présente les résultats de l'apprentissage des réseaux avec les boîtes à moustaches avec la fonction de coût selon les hyperparamètres pour les données de validation échantillonnées à 60s. Fig.4.24.a présente la fonction de coût en fonction des couches cachées, Fig.4.24.b présente la fonction de coût en fonction du nombre de neurones par couche cachée, Fig.4.24.c présente la fonction de coût en fonction du nombre d'epochs et Fig.4.24.d présente la fonction de coût en fonction de l'optimiseur. Il est remarqué un plateau de la fonction de coût avec les hyperparamètres du nombre de couches cachées Fig.4.24.a, du nombre de neurones Fig.4.24.b et du nombre d'epoch, Fig.4.24.c. À l'inverse, il est observé une influence de la sélection de l'optimiseur sur la fonction de coût, Fig.4.24.d. L'optimiseur Oadam engendre une fonction de

coût plus importante que les optimiseurs Adam, Nadam et Radam.

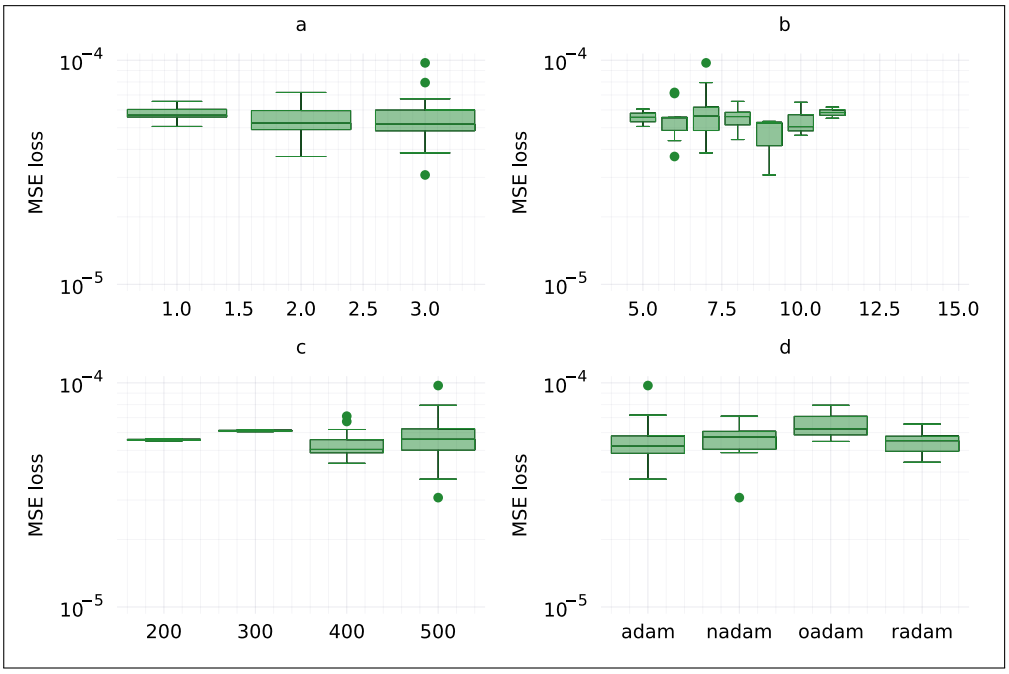

FIGURE 4.24 – boîte à moustache de la fonction de coût avec les données de validation échantillonnées à 60s selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur.

Fig.4.25 présente les résultats de l'apprentissage des réseaux avec les boîtes à moustaches avec la fonction de coût selon les hyperparamètres pour les données de validation échantillonnées à 120s. Fig.4.25.a présente la fonction de coût en fonction des couches cachées, Fig.4.25.b présente la fonction de coût en fonction du nombre de neurones par couche cachée, Fig.4.25.c présente la fonction de coût en fonction du nombre d'epochs et Fig.4.25.d présente la fonction de coût en fonction de l'optimiseur. Il est observé une fonction de coût médiane en plateau selon les quatre hyperparamètres, nombre de couches cachées Fig.4.25.a, nombre de neurones par couches cachées Fig.4.25.b, nombre d'epochs Fig.4.25.c et l'optimiseur Fig.4.25.d.

Fig.4.26 présente les résultats de l'apprentissage des réseaux avec les boîtes à moustaches avec la fonction de coût selon les hyperparamètres pour les données de validation échantillonnées à 300s. Fig.4.26.a présente la fonction de coût en fonction des couches cachées, Fig.4.26.b présente la fonction de coût en fonction du nombre de neurones par couche cachée, Fig.4.26.c présente la fonction de coût en fonction du nombre d'epochs et Fig.4.26.d présente la fonction de coût en fonction de l'optimiseur. Il est observé une fonction de coût médiane en plateau

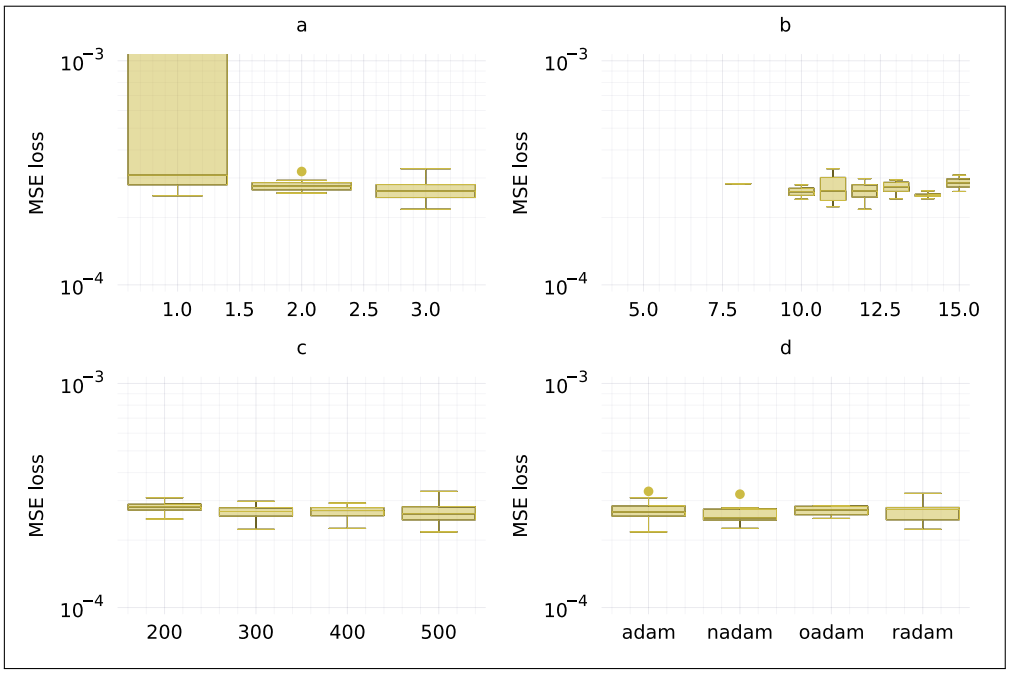

FIGURE 4.25 – boîte à moustache de la fonction de coût avec les données de validation échantillonnées à 120s selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur.

selon les quatre hyperparamètres, nombre de couches cachées Fig.4.26.a, nombre de neurones par couches cachées Fig.4.26.b, nombre d'epochs Fig.4.26.c et l'optimiseur Fig.4.26.d.

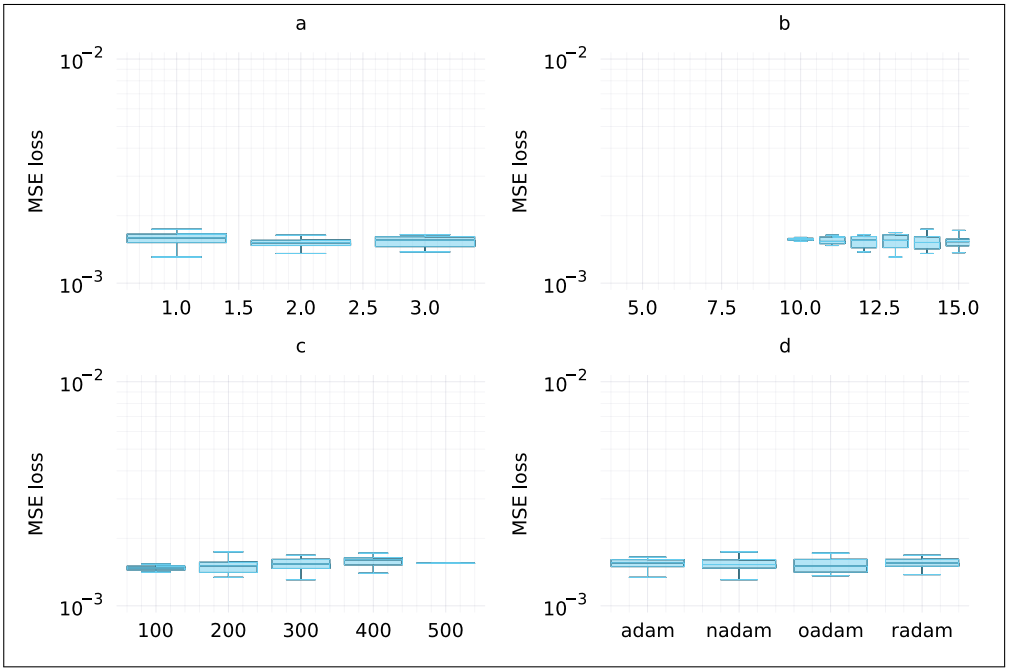

FIGURE 4.26 – boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $300s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur.

Le réseau de neurones sélectionné est le prédicteur réalisant la plus faible MSE avec la période d'échantillonnage de $10s$ et $\mathcal{D}_{10s}$. Les hyperparamètres sélectionnés par l'algorithme metaheuristique sont visibles à la Tab.4.9.

Il est observé une meilleure performance d'identification avec les données de test que les données d'entraînement. Ce résultat est différent des observations réalisées avec le système quatre réservoirs, où l'erreur avec les données de test était plus importante. Une hypothèse est avancée. Une mauvaise sélection des données de test a entrainé un nombre important de données statiques avec les équipements en régime établi ou bien avec un fonctionnement nul. Ensuite, les résultats d'identification observés avec le pas d'échantillonnage de 10 secondes permettent une réduction de l'erreur. Le système dynamique est en apparence plus complexe que le système quatre réservoirs et il y a une très faible augmentation de la fonction de coût (Tab.3.5 pour le système quatre réservoirs et Tab.4.7 pour le système multi-énergie). Deux hypothèses sont avancées pour expliquer une fonction de coût semblable : le système dynamique identifié

TABLE 4.9 – Hyperparamètres minimisants la fonction de coût.

Paramètres	Valeurs
Fonctions d'activations	Relu [195]
Neurones	8
Couches cachées	3
Couche d'intégration pour ResNet	1
Époques	200
Batch size	1024
Optimiseur	Radam [198]
Taux apprentissage	9.71×10^{-4}
Momentum décroissance exponentiel	0.94
Momentum estimation	0.98

est finalement moins complexe qu'envisagé initialement et la fonction de coût atteint l'erreur de discrétisation de l'outil de simulation Dymola [6]. Par ailleurs, le réseau ResNet utilisé dans ce travail possède les mêmes matrices de poids pour chaque couche. Cette structure est utilisée pour obtenir une architecture reliée à la méthode d'intégration de Euler. De plus, elle a aussi été employée pour comparer les performances par rapport au réseau Fnn avec le système quatre réservoirs. Néanmoins, l'utilisation de matrice de poids différents pour chaque couche peut être envisagée pour observer le gain ou la perte de performance pour l'identification du système dynamique.

4.5.2 Pilotage du système multi-énergie

Les résultats des simulations sont présentés par les monotones de puissances pour la chaleur et l'électricité avec une première section pour le régulateur classique et la seconde pour le régulateur avancé. La monotone permet de représenter la durée pendant laquelle la consommation est supérieure à un certain niveau de puissance. Le pas de temps des monotones est la minute et la durée total est de 335 jours correspondant à l'année de production agricole. Ainsi, les abscisses ne correspondent pas au temps de simulation, mais à une durée et les indices de temps sont non consécutifs. Par exemple, une lecture de 50 jours en abscisse et $4.8MW$ en ordonné, indique une consommation supérieure à $4.8MW$ de puissance pendant une durée de 50

6. Une tolérance est renseignée avec le solveur de résolution de Dymola pour le calcul numérique de la dérivée, une valeur de 10^{-6} a été utilisée lors des simulations. Ensuite, extrapoler une finesse finale est difficile car plusieurs paramètres ont une influance, tel que : le solveur utilisé, l'ordre d'intégration ou des paramètres pour accélérer la simulation [237].

jours. Les 50 jours représentant 72000 indices unitaires ne sont pas consécutifs mais répartis dans l'année de production. Ensuite, les sous-figures présentent des variables exogènes (rayonnement, température extérieure, coût de l'électricité) ou du système (température de la serre, état du stockage de la chaleur, débit d'injection du CO_2, la commande de l'unité de co-génération, la commande de la chaudière) avec d'iso-contour de densité de probabilité des valeurs de la données en ordonnées. Ces sous-figures sont synchronisées en abscisse avec la monotone de puissance de la première figure. Par conséquent, les sous-figures présentent toujours une durée en abscisse. Les couleurs froides (bleu) indiquent une probabilié d'occurrences faible et les couleurs chaudes (rouge) indiquent une probabilité d'occurrences élevé. L'estimation de la densité de probabilité est réalisées avec l'estimation par noyau (*kernel density estimation* en anglais) [238] :

$$\hat{f}_{kde}(x) = \frac{1}{hN} \sum_{j=1}^{N} K\left(\frac{x - X_j}{h}\right) \tag{4.57}$$

avec, h la fenêtre (*bandwidth*), N le nombre d'échantillons, $K(x)$ le noyau et X_j les échantillons. Le calcul a été réalisé avec la boîte à outils *KernelDensity.jl* [239] utilisant la méthode de Silverman pour le calcul de h et la fonction normale (ou Gaussienne) pour $K(x)$ (options par défaut de la boîte à outils).

La Fig.4.27 présente les résultats de la simulation du pilotage avec le régulateur classique avec la monotone de puissance thermique, puis la Fig.4.28 avec la monotone de puissance électrique. De plus, la Fig.4.30 présente les résultats du pilotage avec le régulateur avancé avec la monotone de puissance thermique, puis la Fig.4.31 avec la monotone de puissance électrique. En outre, afin de présenter la densité de probabilité pour Fig.4.27.d, Fig.4.27.e, Fig.4.27.i, Fig.4.28.b et Fig.4.30.d, Fig.4.30.e, Fig.4.30.i, Fig.4.31.b un traitement des données a été réalisé. En effet, les échantillons des débits de $CO2$ et de rayonnement nuls ont été supprimés avant le calcul par l'équation (4.57).

Afin de faciliter l'analyse et la présentation des résultats, les figures sont divisées en période d'étude temporelle, celles-ci sont indiquées par les flèches jaunes sur les Fig.4.27, Fig.4.30, Fig.4.28 et Fig.4.31. Nous pouvons remarquer que les comportements des deux régulateurs diffèrent très sensiblement selon les zones considérées. Notons que pour ces zones, les comparaisons entre les deux régulateurs sont limitées par le fait qu'on ne considère pas les mêmes instants, car les rangements temporels des monotones de puissance diffèrent. Il est observé que la puissance de la serre agricole (courbe orange) est lisse alors que les puissances du stockage thermique, du réseau basse température de l'unité de co-génération et de la chaudière ne sont

pas lisses avec des dépassements de puissance, Fig.4.27.a et Fig.4.30.a.

Les puissances présentées sont échantillonnées à la minute et il a été observé quelques échantillons dont les puissances sont négatives, en particulier sur le réseau basse température de la chaudière et de l'unité de co-génération. Ces échantillons de puissances négatives minoritaires sont filtrés pour l'affichage de la monotone, néanmoins elles ont été prises en compte pour le bilan thermique de la serre agricole.

Régulateur classique

Nous pouvons présenter les périodes d'analyses du régulateur classique pour le vecteur chaleur avec la Fig.4.27 :

— Première période (0 à 37.5 jours) : Les puissances de chauffage transmises à la serre agricole sont les plus élevées, avec des maxima de $11.3 MW$ de faible durée. La puissance est transmise à la serre principalement par le stockage thermique et secondairement par la chaudière et l'unité de co-génération par le réseau base température, Fig.4.27.a. Sur un ensemble d'indices temporels équivalent aux 9 premiers jours de la monotone de puissance, les iso-contours d'occurrence de la température interne de la serre montre la récurrence importante de valeurs situées entre $20°C$ et $22°C$, Fig.4.27.b. Ces indices temporels coïncident certainement avec des périodes de changement du point de consigne de la serre agricole de $20°C$ à $22°C$ en mi-saison sans fonctionnement de l'unité de co-génération, Fig.4.27.f. Les pics de consommations en cette période de l'année, plutôt qu'au cœur hivernale, semble contre-intuitif, mais pourrait être expliqués par l'utilisation ou non de l'écran thermique pendant ces périodes. Pour les indices temporels après 9 jours, les iso-contours présentent des occurences de rayonnement solaire inférieur à $250 Wm^{-2}$, Fig.4.27.i, et de températures extérieures à la serre faibles, le plus souvent inférieures à $10°C$, Fig.4.27.h. Ainsi, cette première période correspond à des instants de chauffage de la serre agricole intense. Ces intants surviennent en période d'utilisation de l'unité de co-génration (Fig.4.27.f), entre 6h et 21h, quand le soleil est levé, avec une consigne de température de la serre de $22°C$, incluant aussi la période de transition de $20°C$ à $22°C$ de la température de la serre. Les iso-contours de densité de probabilité montrent que le SOC prend fréquemment des valeurs entre 0.1 et 0.6, avec une prédominance à 0.4. Nous pouvons en déduire que l'état de charge du stockage thermique varie entre 10% et 60%, et plus régulièrement autour de 40%. Le stockage thermique est donc sollicité : il se décharge puis est rechargé par les unités de production, Fig.4.27.c, Fig.4.27.f et Fig.4.27.g. Les iso-contours de densité de probabilité montrent que le CO_2 prend majoritairement

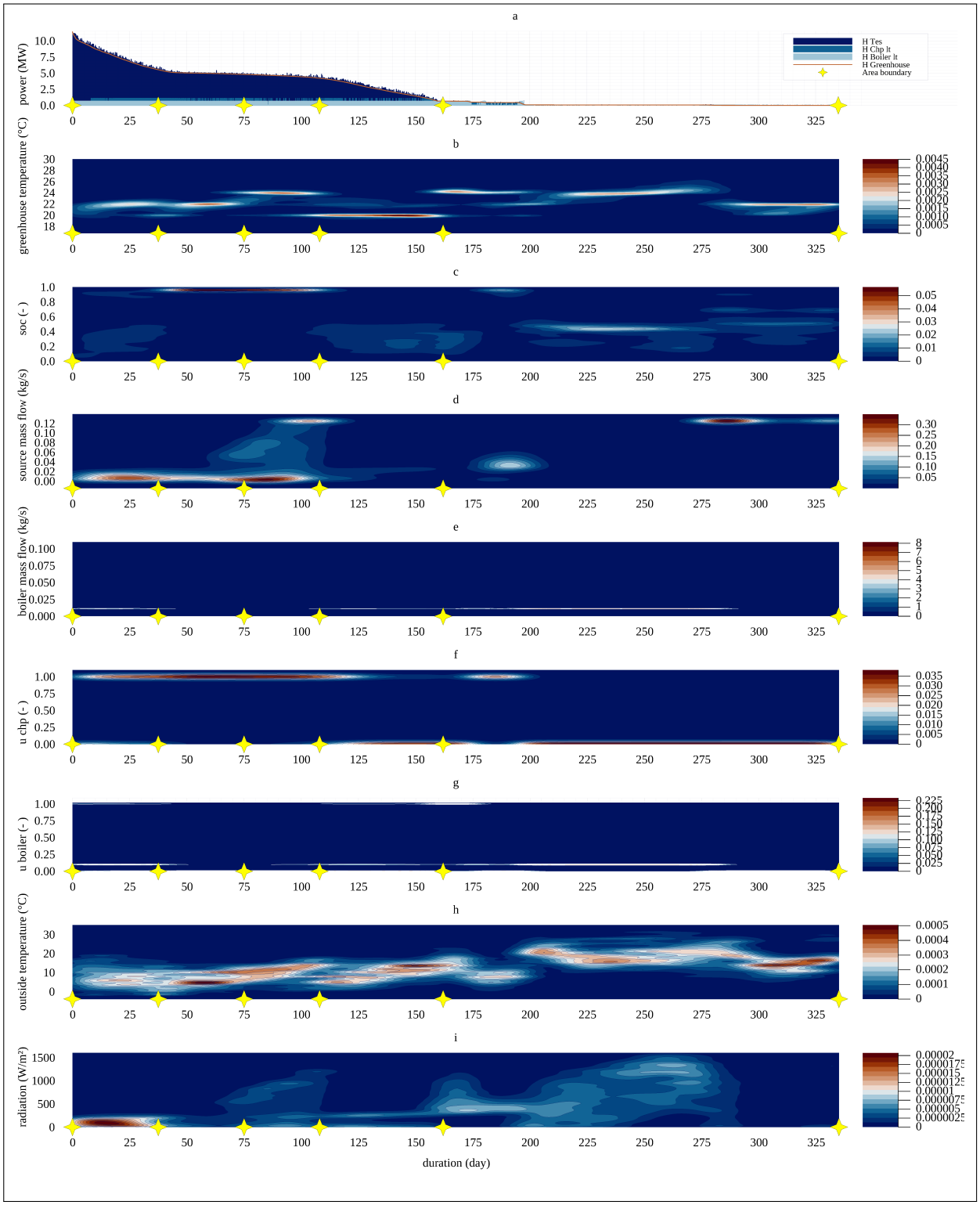

FIGURE 4.27 – Pilotage avec le régulateur classique, (a) monotones de puissance thermique, (b) la température de la serre, (c) l'état de charge du stockage, (d) le débit de CO_2 de la source, (e) le débit de CO_2 de la chaudière, (f) la commande de l'unité de co-génération, (g) la commande de la chaudière, (h) la température extérieure, (i) le rayonnement.

une valeur de $0.01 kg s^{-1}$ pour la source de CO_2 et de $0.0125 kg s^{-1}$ pour la chaudière, Fig.4.27.d et Fig.4.27.e. Le vecteur fertilisant de la serre agricole est fourni par la source de CO_2 et de la valorisation des gaz d'échappement de la chaudière.

— Deuxième période (37.5 à 75 jours) : Les puissances de chauffage transmises à la serre agricole sont plus faibles que lors de la première période, Fig.4.27.a. Nous sommes sur la première partie du plateau autour de $4.8 MW$ (entre $5.1 MW$ et $4.4 MW$), l'unité de co-génération reste en fonctionnement, Fig.4.27.f ainsi que la chaudière Fig.4.27.g. Les températures sont inférieures à $10°C$ et le rayonnement solaire est faible (les occurences les plus présentes sont entre 0 et $300 W m^{-2}$), Fig.4.27.h et Fig.4.27.i. Nous sommes certainement en début et fin de journée (entre 6h et 21h, autour des levers et couchers du soleil, et après le changement de consigne), et en période de chauffe pour maintenir la consigne de température à $22°C$ dans la serre agricole, Fig.4.27.b. Le SOC du stockage thermique est maintenu à 1 par le fonctionnement conjugué de l'unité de co-génération et de la chaudière, et malgré des sollicitations importantes, Fig.4.27.c.

— Troisième période (75 à 108 jours) : Les températures extérieures sont plus élevées que la deuxième zone (entre $8°C$ et $20°C$), Fig.4.27.h. En outre, le rayonnement solaire est variable, de 0 à $1100 W m^{-2}$, Fig.4.27.i. A priori, dans ces conditions, on devrait moins solliciter la chaleur du stockage thermique. Pourtant, il y a la présence de plateaux pour la puissance de la demande de chaleur à $4.8 MW$, pour l'état de charge du stockage à 1 et pour la température de la serre agricole à $24°C$ (supérieure de $2°C$ à la consigne), Fig.4.27.a, Fig.4.27.b. et Fig.4.27.c. De plus, les iso-contours de densité de la source de CO_2 sont variables avec des débits allant de $0.02 kg s^{-1}$ à $0.12 kg s^{-1}$.

Pour les deuxième et troisième périodes, l'unité de co-génération fonctionne à son régime nominal et la chaudière fournie parfois de la chaleur alors que l'état de charge du stockage thermique est proche de 1. Il y a donc dans les meilleurs cas un équilibre entre la chaleur produite et fournie au stockage et la chaleur transmise à la serre agricole. Dans ces cas, la consigne de température dans la serre est respectée (deuxième période). Dans les cas les moins favorables, plus de chaleur est produite par l'unité de co-génération et la chaudière que la demande nécessaire pour maintenir la consigne de température de la serre. Dans ces cas, l'excès de chaleur est dissipé dans la serre, le stockage étant déjà saturé, ainsi la consigne de température n'y est plus respectée (troisième période). Dans tous les cas, l'intérêt du stockage est limité puisque l'énergie ne fait qu'y transiter. En outre, la dissipation de la chaleur dans la serre occasionne la nécessité de la refroidir pour maintenir la température à maximum $2°C$ de la consigne. Le refroidissement est

réalisé par le renouvellement d'air frais. Cet air frais est aussi moins concentré en CO_2 et un apport supplémentaire de CO_2 est nécessaire pour maintenir la concentration à la consigne. À ce titre, le système énergétique ne semble pas optimisé.

— Quatrième période (108 à 162 jours) : Les puissances thermiques transmises à la serre ne sont plus en plateau, mais elles se réduisent linéairement, Fig.4.27.a. La quatrième période présente deux régimes dictincts de température dans la serre. Le premier régime contient les occurrences de température de la serre agricole autour de la consigne de $20°C$, Fig.4.27.b. La serre agricole est dans une période avant 6h00 ou après 21h correspondant avec des occurrences de rayonnement solaire inférieur à $200 W m^{-2}$ voire proche de 0, Fig.4.27.h et Fig.4.27.i. Le second régime contient les occurrences de température de la serre agricole autour de la consigne de $22°C$, Fig.4.27.b. La serre agricole est dans une période après 6h00 et avant 21h correspondant aux occurrences de rayonnement solaire autour de $200 W m^{-2}$ et $300 W m^{-2}$, Fig.4.27.i. Nous pouvons observer que les valeurs de températures extérieures récurrentes augmentent à mesure de l'avancement de cette période, passant de $10°C$ à environ $20°C$ à la fin de la période. Conformément, les demandes et la chaleur fournie diminue logiquement, Fig.4.27.h. Le SOC présente des occurrences variables entre 0.1 et 0.4, Fig.4.27.c. Par ailleurs, de 108 à 125 jours, l'unité de co-génération fonctionne le plus souvent à puissance nominale, alors que la puissance thermique fournie à la serre est encore élevée. Tandis qu'elle n'est pas en fonctionnement de 125 à 162 jours, quand moins de chaleur est fournie à la serre, Fig.4.27.f. En parrallèle à ces instants, la chaudière connait un fonctionnement inverse : peu sollicitée sinon pour produire du CO_2 entre 108 et 125 jours, elle est utilisée régulièrement à régime nominal de 125 à 162 jours, Fig.4.27.g.

— Cinquième période (162 à 335 jours) : La serre agricole ne demande pas ou peu de puissance de chauffage, et les puissances transmises faibles résultent du réseau basse température, Fig.4.27.a. Les iso-contours de densité de probabilité montrent que les températures extérieures sont le plus souvent au-delà de $10°C$, un rayonnement solaire variable, entre 0 et $1500 W m^{-2}$, et des températures de la serre agricole de $20°C$ à $24°C$, Fig.4.27.h, Fig.4.27.i. Fig.4.27.b. En outre, les iso-contours montrent que la source de CO_2 ainsi que la chaudière fournissent du fertilisant à la serre agricole sur cette période, Fig.4.27.d et Fig.4.27.e. Concernant la source de CO_2, les iso-contours sont concentrés sur deux périodes : la première période de 175 à 200 avec $0.04 kg s^{-1}$; la seconde période de 275 à 335 avec $0.012 kg s^{-1}$. Pour la chaudière, les iso-contours montrent une densité de débit de $0.0125 kg s^{-1}$ avec une période de 175 à 287. La serre agricole est en dehors de la période

de chauffe, car pas ou peu de chauffage est sollicité et du $CO2$ est consommé. Cette zone correspond à un fonctionnement de la serre lors de périodes pendant lesquelles la serre n'a pas besoin d'être chauffée, voire un besoin de refroidissement. Ces indices temporels sont dispersés sur l'année, et existent même en période de chauffe, quand les conditions ambiantes le permettent. Elles sont cependant plus fréquentes en été, période pendant lesquels des journées entières sont sous ce régime.

Nous pouvons présenter les périodes d'analyses du régulateur classique pour le vecteur électrique avec la Fig.4.28.

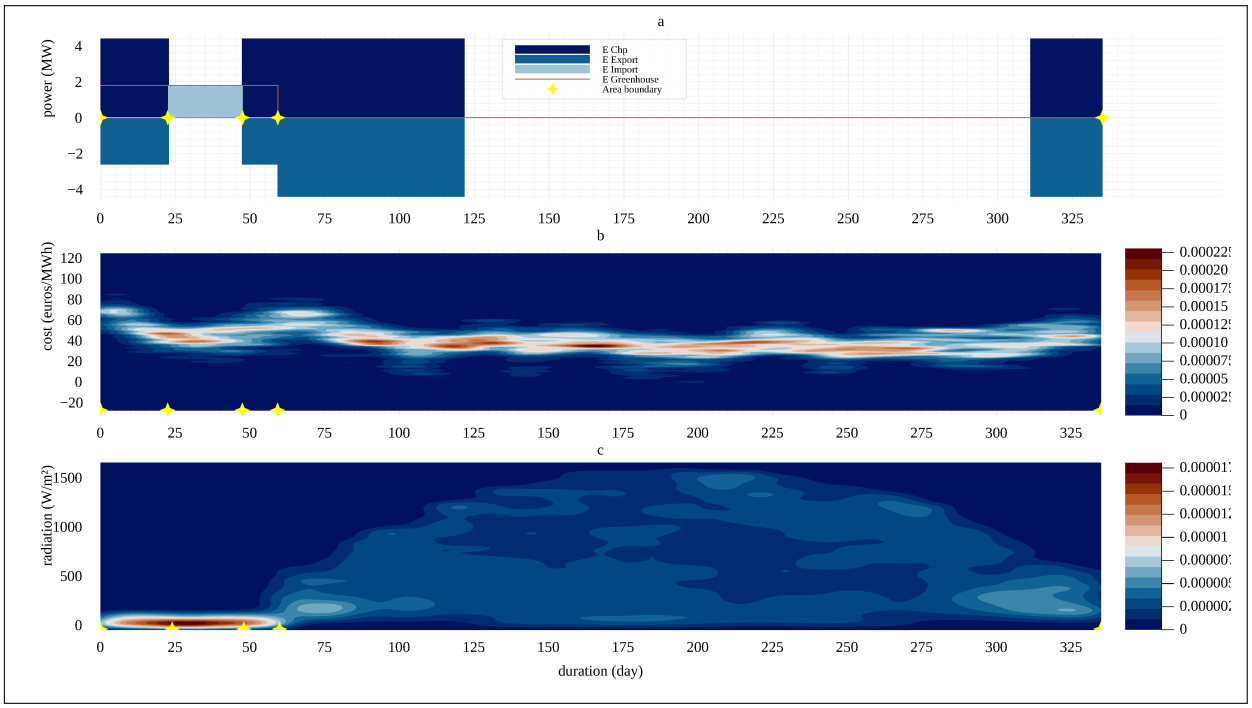

FIGURE 4.28 – Pilotage avec le régulateur classique, (a) monotones de puissance électrique, (b) le coût de l'électricité, (c) le rayonnement.

— Première (0 à 22.5 jours) et troisième périodes (47.5 à 59.3) : Nous sommes en présence de demande électrique de la serre agricole, Fig.4.28.a. Il est observé que les iso-contours de densité de probabilité montrent des récurrences prédominantes de valeurs de rayonnement inférieure à $100\,W\,m^{-2}$, Fig.4.28.c. La serre agricole demande de l'électricité lorsque le rayonnement solaire est faible pour démarrer les lampes et augmenter le rayonnement reçu par les plantes. La demande électrique de la serre temporellement est visible à la Fig.4.29. Ce démarrage des lampes subvient en début ou la fin d'une journée. En outre, durant cette période, l'unité de co-génération fonctionne, la demande électrique est plus

faible que la production et une partie de d'électricité est exportée au réseau, Fig.4.28.a. Cette période correspond aux mois entre novembre et mars avec le début ou la fin de la journée. Les coûts électriques sont les plus importants avec des iso-contours entre 50 à 70 €/MWh, Fig.4.28.b.

— Deuxième période (22.5 à 47.5 jours) : Nous sommes en présence de demande électrique de la serre agricole, Fig.4.28.a. Il est observé que les iso-contours du rayonnement sont inférieure à 100 Wm^{-2}, Fig.4.28.c. Durant la deuxième période, l'unité de co-génération de produit pas d'électricité et toute la demande électrique est fournie par le réseau électrique. Cette période correspond aux mois entre mars et novembre, quand l'unité de co-génération de fonctionne pas, Fig.4.28.a. En outre, les iso-contours de densité du coût électrique varient entre 20 et 60 €/MWh, Fig.4.28.b. Ce tarif est plus faible par rapport au périodes 2 et 4.

— Quatrième période (59.3 à 335 jours) : La demande électrique est nulle et deux régimes de fonctionnement de l'unité de co-génération sont observables, Fig.4.28.a. Le premier est la production électrique et la vente et le second est une production électrique nulle. Les iso-contours du rayonnement solaire sont entre 100 Wm^{-2} et 1500 Wm^{-2}, Fig.4.28.b. Les iso-contours de densité du coût électrique varient entre un coût très faible de 0 à 70 €/MWh, Fig.4.28.b. Cette période correspond à la journée de production agricole tout au long de l'année.

Régulateur avancé

Nous pouvons aborder les périodes d'analyses du régulateur avancé pour le vecteur chaleur avec la Fig.4.30 :

— Première période (0 à 37.5 jours) : Cette première période présente des similitudes avec la première période du régulateur classique avec les puissances de chauffage, les iso-contours de la température de la serre, du rayonnement et de la température extérieure comparable, et en conséquence une analyse similaire. Néanmoins, les 9 premiers jours de la période présente un apport de chaleur pour la serre avec le stockage thermique, le réseau basse température de la chaudière et l'unité de co-génération, Fig.4.30.a. Ceci présente une distinction par rapport au régulateur classique, car l'unité de co-génération ne fournit pas de chaleur. Ensuite, les iso-contours de densité de probabilité montrent que le SOC prend fréquemment des valeurs entre 0.1 et 0.8, avec une prédominance à 0.4. Nous pouvons en déduire que l'état de charge du stockage thermique varie entre 10% et 80%, et plus régulièrement autour de 40%. Le stockage thermique est donc sollicité :

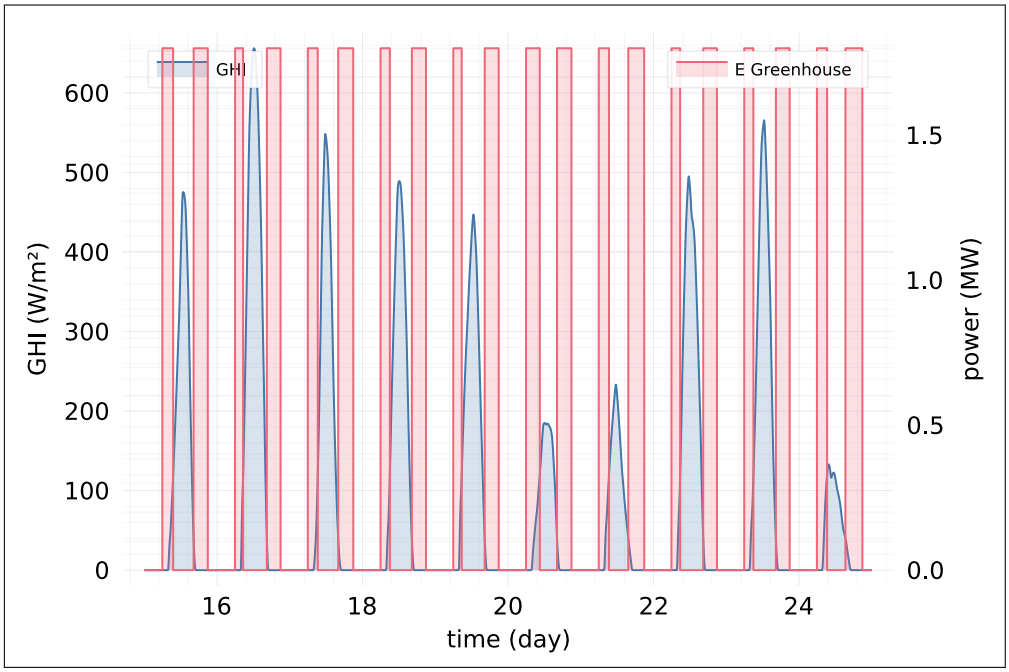

FIGURE 4.29 – Série temporelle du rayonnement solaire et de la puissance électrique.

il se décharge puis est rechargé par les unités de production, Fig.4.30.c, Fig.4.30.f et Fig.4.30.g. Les iso-contours de densité de probabilité montrent que le CO_2 prend des valeurs entre $0.01 kg s^{-1}$ et $0.02 kg s^{-1}$ pour la source de CO_2 et de $0.0125 kg s^{-1}$ pour la chaudière, Fig.4.30.d et Fig.4.30.e. Le vecteur fertilisant de la serre agricole est fourni par la source de CO_2 et de la valorisation des gaz d'échappement de la chaudière.

— Deuxième période (37.5 à 100 jours) : Les puissances de chauffage transmises à la serre agricole sont plus faibles que lors de la première zone et varient quasi linéairement, Fig.4.30.a. Les iso-contours de la température de la serre montrent qu'il existe beaucoup d'occurrence pendant lesquels la serre est maintenue à $20°C$, montrant que la majorité des instants de cette période sont soit avant 6h ou après 21h, Fig.4.30.b. Cependant, nous pouvons observer qu'il existe des occurrences pendant lesquelles la température de la serre est maintenue à $22°C$, situant ces instants entre 6h et 21h, Fig.4.30.b. Il s'agit d'instants en journée pendant lesquels les températures sont relativement basses (sous $15°C$ selon les iso-contours de densité de probabilité des températures extérieures, Fig.4.30.h) et le rayonnement solaire solaire est faible (inférieur à $200 W m^{-2}$ et augmentant linéairement, selon les iso-contours de densité de probabilité, Fig.4.30.i). La chaleur fournie évolue ici avec les conditions environnementales et les consignes. Le stockage de chaleur est sollicité avec un SOC entre 0.2 et 0.6, Fig.4.30.c. L'unité de co-génération

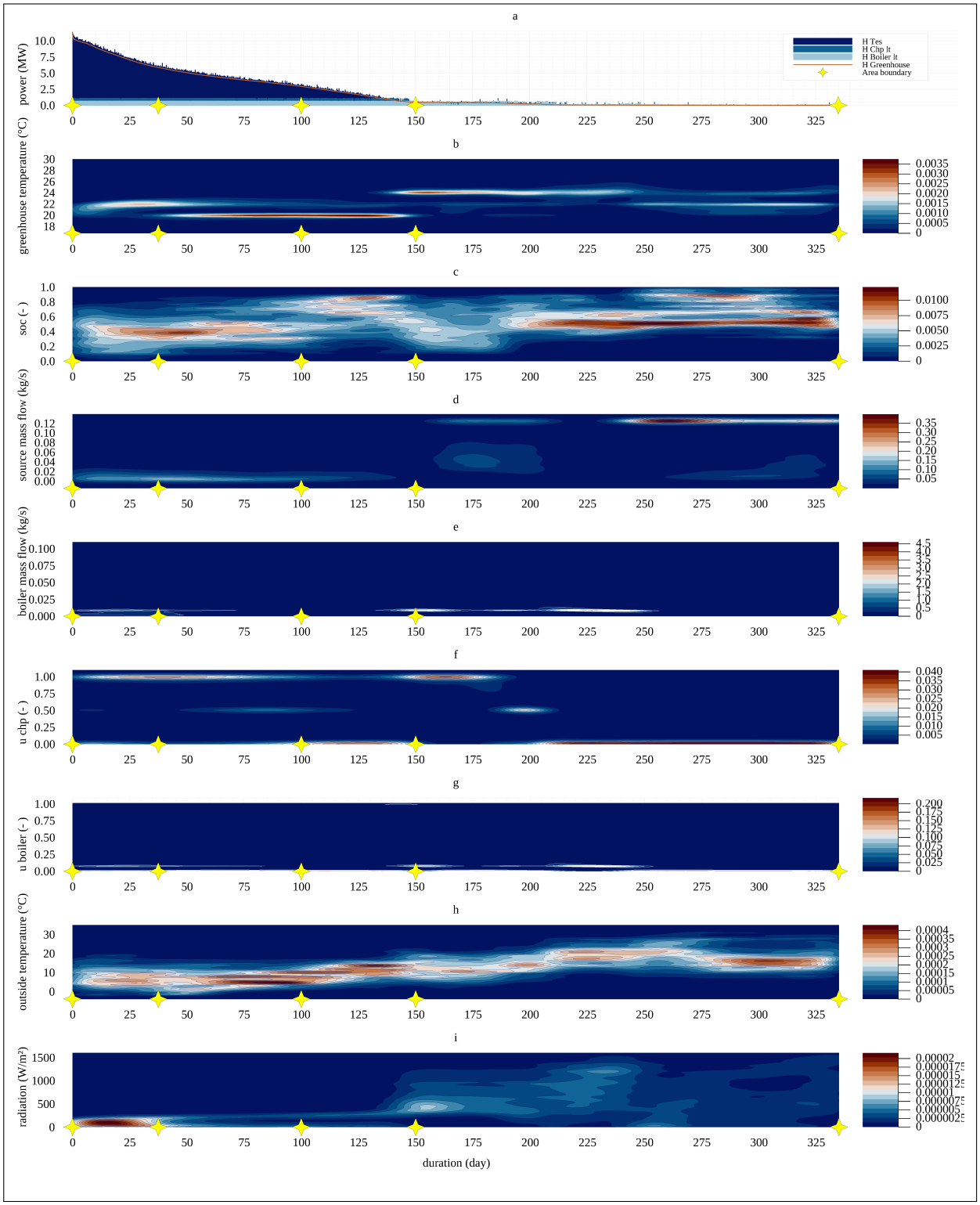

FIGURE 4.30 – Pilotage avec le régulateur avancé, (a) monotones de puissance thermique, (b) la température de la serre, (c) l'état de charge du stockage, (d) le débit de CO_2 de la source, (e) le débit de CO_2 de la chaudière, (f) la commande de l'unité de co-génération, (g) la commande de la chaudière, (h) la température extérieure, (i) le rayonnement.

fonctionne pour fournir la puissance de chauffage avec soit la puissance maximale ou la demi-puissance et la chaudière ne fonctionne pas, Fig.4.30.f et Fig.4.30.g. Nous pouvons en conclure qu'il y a un ordre de priorité entre l'unité de co-génération et la chaudière établi certainement par le fait que la revente d'électricité équilibre le coût du gaz par l'unité de co-générartion. Ainsi, l'apport de CO_2 à la serre est fourni par la source, Fig.4.30.d.

— Troisième période (100 à 150 jours) : Comme pour la seconde période, les iso-contours de températures de la serre montrent qu'il existe beaucoup d'occurrence pendant lesquelles la serre est maintenue à $20°C$. Ces périodes coïncident avec des instants avant 6h ou après 21h. Néanmoins, il y a aussi des occurrences pendant lesquelles la température de la serre est maintenue à $22°C$, situant ces instants entre 6h et 21h, Fig.4.27.b. Cependant, pendant ces instants, la serre subit des températures extérieures plus élevées (jusqu'à $20°C$) et reçoit plus de rayonnement solaire (250 à $400\,Wm^{-2}$), Fig.4.27.h et Fig.4.27.i. Il est donc normal que dans ces conditions, les puissances thermiques injectées dans la serre soient inférieures à la seconde période, Fig.4.27.a. La plupart du temps, ni l'unité de co-génération ni la chaudière ne fonctionnent dans la troisième période, Fig.4.27.f et Fig.4.27.g. Le stockage thermique est modérément sollicité, et en conséquence, le SOC à tendance à rester élevé (au-delà de 0.4 et régulièrement proche de 1), Fig.4.27.c. Le stockage chargé pendant des instants d'autres périodes fournit la chaleur nécessaire à la serre.

— Quatrième période (150 à 335 jours) : Tout comme le régulateur classique, la serre agricole ne demande pas de puissance de chauffage, et les puissances transmises faibles résultent du réseau basse température, Fig.4.30.a. Les iso-contours de densité de probabilité montrent que les températures extérieures sont le plus souvent au-delà de $10°C$, un rayonnement solaire variable, entre 0 et $1500\,Wm^{-2}$, et des températures de la serre agricole de $20°C$ à $24°C$, Fig.4.30.h, Fig.4.30.i. Fig.4.30.b. En outre, les iso-contours montrent que la source de CO_2 ainsi que la chaudière fournissent du fertilisant à la serre agricole sur cette période, Fig.4.30.d et Fig.4.30.e. Concernant la source de CO_2, les iso-contours sont concentrés sur deux périodes : la première période de 150 à 250 avec des débits variant de $0kgs^{-1}$ à $0.012kgs^{-1}$, puis la seconde période de 250 à 335 avec une pré-dominance de fort débit à $0.012kgs^{-1}$. Pour la chaudière, les iso-contours montrent une densité de débit de $0.0125kgs^{-1}$ avec une période de 150 à 250, puis une absence d'iso-contours de 250 à 335. Ici, nous en déduisons que le CO_2 en provenance de la chaudière est fournie sur une période un peu plus longue avec un débit moindre

alors que la source fournie le CO_2 sur une période réduite avec un débit plus important. La serre agricole est en dehors de la période de chauffe, car pas ou peu de chauffage est sollicité et du $CO2$ est consommé. Cette zone correspond à un fonctionnement de la serre lors de périodes pendant lesquelles la serre n'a pas besoin d'être chauffée, voire un besoin de refroidissement. Ces indices temporels sont dispersés sur l'année, et existent même en période de chauffe, quand les conditions ambiantes le permettent. Elles sont cependant plus fréquentes en été, période pendant lesquels des journées entières sont sous ce régime.

Nous pouvons présenter les périodes d'analyses du régulateur avancé pour le vecteur électrique avec la Fig.4.31.

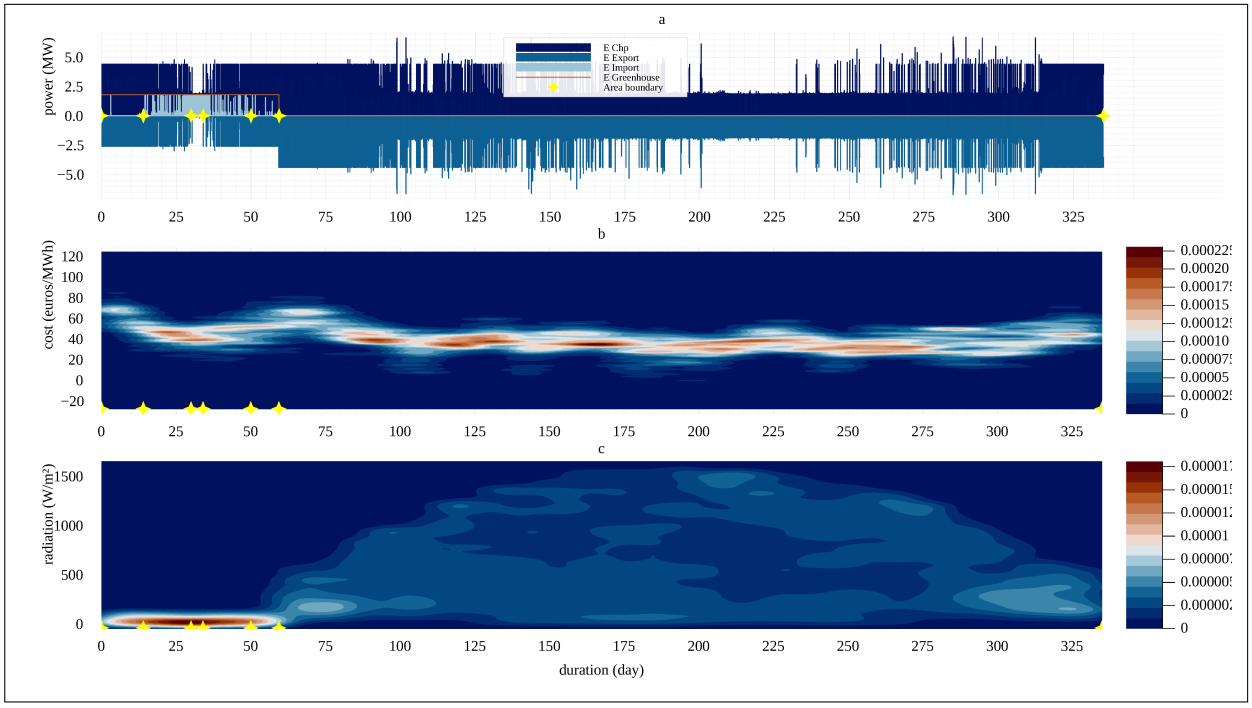

FIGURE 4.31 – Pilotage avec le régulateur avancé, (a) monotones de puissance électrique, (b) le coût de l'électricité, (c) le rayonnement.

— Première (0 à 14 jours) et cinquième période (50 à 59.4 jours) : Comme pour le régulateur classique, nous sommes en présence de demande électrique de la serre agricole, Fig.4.31.a. Il est observé que les iso-contours du rayonnement sont inférieure à $100\,W\,m^{-2}$, Fig.4.31.c. La serre agricole demande de l'électricité lorsque le rayonnement solaire est faible pour démarrer les lampes et augmenter le rayonnement reçu par les plantes. La demande électrique de la serre temporellement est visible à la Fig.4.29. Ce démarrage

des lampes se présente en début ou la fin d'une journée. En outre, les iso-contours du coût électrique varient de 50 à 70 €/MWh, Fig.4.31.b. Durant cette période, l'unité de co-génération fonctionne, la demande électrique est plus faible que la production et une partie de d'électricité est exportée au réseau, Fig.4.31.a.

— Deuxième (14 à 30 jours) et quatrième périodes (34 à 50 jours) : La demande électrique de la serre est soutenue, Fig.4.31.a et comme le montre les iso-contours du rayonnement solaire l'apport lumineux naturel est inférieure à 100 Wm^{-2}, Fig.4.31.c. Les lampes sont allumées, l'unité de co-génération alterne entre un régime de fonctionnement et d'arrêt, Fig.4.31.a. Lorsque l'unité est arrétée, l'énergie électrique est soutirée du réseau. Lorsque l'unité fonctionne, l'énergie électrique produite est utilisée pour la serre agricole et le surplus est exporté sur le réseau.

— Troisième période (30 à 34 jours) : La demande électrique de la serre est soutenue, Fig.4.31.a et comme le montre les iso-contours du rayonnement solaire l'apport lumineux naturel est inférieure à 100 Wm^{-2}, Fig.4.31.c. Les lampes sont allumées et l'unité de co-génération ne produit pas d'électricité ainsi toute la demande électrique est fournie par le réseau, Fig.4.31.a. Les iso-contours du coût électrique sont au plus bas (entre 20 et 60 €/MWh) et malgré la demande électrique, l'unité de co-génération n'est pas démarée, Fig.4.31.b. L'hypohèse formulée est que la demande de chaleur est réduite et qu'il n'est pas possible de valoriser cette production. Ainsi, l'énergie électrique est soutirée au réseau et la chaleur est fournie par le stockage et/ou la chaudière.

— Sixième période (59.4 à 335 jours) : La demande électrique est nulle et les iso-contours du rayonnement solaire sont entre 100 Wm^{-2} et 1500 Wm^{-2}, Fig.4.31.a. et Fig.4.31.c. L'unité de co-génération présente trois régimes de fonctionnement, un fonctionnement à la puissance maximale, un fonctionnement à puissance réduite et un arrêt de l'unité de co-génération, Fig.4.31.a. Cette période se présente en journée entre 6h et 21h tout au long de l'année. Ici, nous pouvons faire l'hypothèse que le fonctionnement de l'unité de co-génération est conditionné au besoin de chaleur de la serre agricole.

Comparaisons

Nous pouvons comparer les différences de fonctionnement entre ces deux régulateurs, en particulier sur les périodes 2 et 3 du régulateur classique et la période 2 du régulateur avancé avec les figures des monotones de chaleur, Fig.4.27.a et Fig.4.30.a. Dans le premier cas, le fonctionnement des systèmes n'est pas énergétiquement optimisé : l'unité de co-génération fonctionne par défaut, la chaudière est sollicitée pour produire du CO_2 et en conséquence, le

SOC du stockage plafonne à 1 alors que les températures dans la serre dépassent régulièrement les consignes montrant qu'un surplus de chaleur transitant dans le stockage est alors dissipé dans la serre. Dans le second cas, les systèmes sont énergétiquement optimisés : ils sont utilisés quand nécessaire pour recharger le stockage, le SOC varie entre 0.4 et 0.6, alors que les températures de consigne sont respectées, montrant que les flux de chaleur dans le stockage sont bien régulés, puis la source est sollicité pour fournir le CO_2 à la serre agricole.

Des synergies entre vecteurs sont observées grâce au fonctionnement de la chaudière avec la production de chaleur et le CO_2. En effet, pour les deux régulateurs, la première période présente le fonctionnement de la chaudière pour la consommation de chaleur et de CO_2 par la serre. Ensuite, une autre synergie est observable avec la dernière période pour les deux régulateurs (4 et 5). Durant cette période, la chaudière fonctionne en journée pour produire du CO_2 et la chaleur non utilisée est stockée. Ensuite, la chaleur produite durant la journée est consommée la nuit (période 3 pour le régulateur avancé et la période 4 pour le régulateur classique), Fig.4.27.c, Fig.4.30.c et Fig.4.27.e, Fig.4.30.e.

Nous pouvons ensuite comparer le fonctionnement de l'unité de co-génération par rapport au coût électrique avec sa monotone, Fig.4.32.a. Tout d'abord, les iso-contours de la demande électrique de la serre sont concentrés entre la période de 0 à 175, Fig.4.32.d. En outre, les iso-contours de la demande de chaleur sont les plus importants entre la période de 0 à 75, Fig.4.32.e et Fig.4.32.f. Ces périodes correspondent à la tarification la plus élevée, Fig.4.32.a. Les iso-contours de la commande de l'unité de co-génération sont étalés sur les 335 jours avec le régulateur classique (avec une forte concentration de 0 à 50 jours), Fig.4.32.b. L'unité fonctionne quelque soit la tarification électrique, ce qui coïncide avec la règle d'utilisation de l'unité de co-génération, indépendamment des coûts d'électricité avec ce régulateur, Fig.4.32.c. Par ailleurs, nous pouvons noter que la demande de chaleur de la serre n'est pas nulle pour les derniers instants de la monotone, des iso-contours montrent une demande chaleur variant de 0 à $3MW$, Fig.4.32.f. Ainsi, malgré une demande de chaleur, on utilise de moins en moins l'unité de co-génération quand les coûts de l'électricité baissent, Fig.4.32.a et Fig.4.32.c. Le régulateur avancé est optimisé sur le coût de l'énergie, entre autres, considérant les recettes de vente de l'électricité. Lorsque le tarif de vente d'électricité (équivalent au coût) est élevé, celui-ci équilibre le coût du gaz alimentant l'unité de co-génération. Il est alors intéressant d'utiliser ce système énergétique. Dans le cas contraire, l'unité devient moins compétitive par rapport à la chaudière dont l'efficacité de conversion du gaz en chaleur est plus importante.

Suite aux comparaisons, les bilans énergétiques et économiques sont abordés. La Tab.4.10 présente l'énergie produite et échangée ainsi que les consommations d'énergie de gaz naturel

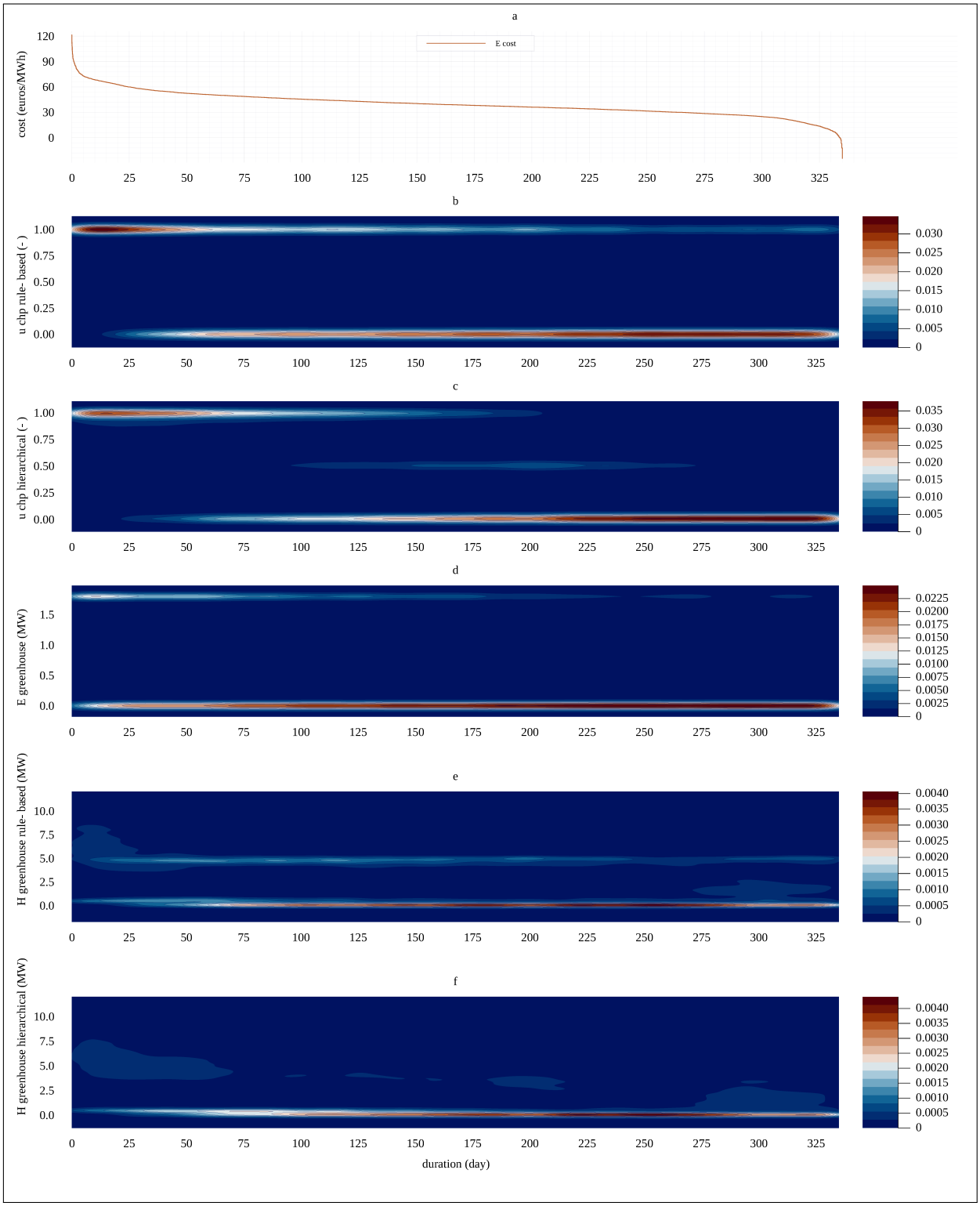

FIGURE 4.32 – Pilotage avec le régulateur classique et avancé, (a) monotone du coût électrique, (b) la commande par le régulateur classique de l'unité de co-génération, (c) la commande par le régulateur avancé de l'unité de co-génération, (d) la demande électrique, (e) la demande de chaleur de la serre avec le régulateur classique, (f) la demande de chaleur de la serre avec le régulateur avancé.

et de CO_2 pour les deux simulations de l'étude. Tout d'abord, la chaleur consommée par la serre avec le régulateur classique est de 312 MWh, convernant le régulateur avancée la consommation est de 270 MWh. La baisse de la consommation est de 13.5%. Ensuite, la chaleur produite par la chaudière est plus importante avec le régulateur classique que le régulateur avancé, il est observé une production de 99.55MWh par rapport à 62.23MWh. Ensuite, l'énergie thermique et électrique produite par l'unité de co-génération est légèrement inférieure pour le régulateur avancé que le régulateur classique, avec 208.07MWh pour 212.88MWh. Cet élément est à mettre en perspective avec une utilisation de l'unité de co-génération étalée pour le régulateur avancé (Fig.4.31.a) et concentré pour le régulateur classique (Fig.4.28.a). Par la suite, l'énergie électrique exportée est de 187.97 MWh pour le régulateur classique et 181.24 MWh pour le régulateur avancé. L'énergie importée est de 17.80 MWh pour le régulateur classique et 15.88 MWh pour le régulateur avancé. Le régulateur avancé présente un échange avec le réseau électrique moindre, que le régulateur classique. Ensuite, la consommation de gaz naturel est plus faible avec le régulateur avancé que le régulateur classique avec 76.03 T par rapport à 80.79 T. Finalement, la consommation de CO_2 est plus importante avec le régulateur classique que le régulateur avancé avec 27.22 T pour 24.12 T. La répartition pour le régulateur de règles expertes est de 4.97 T de CO_2 en provenance de la source et 22.25 T en provenance de la chaudière. Concernant le régulateur hiérarchique, 10.21 T proviennent de la source et 13.91 T de la chaudière.

Table 4.10 – Bilan énergétique des simulations.

	Régulateur classique	Régulateur avancé
Énergie thermique consommée par la serre	312 MWh	270 MWh
Énergie thermique produite par la chaudière	99.55 MWh	62.23 MWh
Énergie thermique produite par la chp	212.88 MWh	208.07 MWh
Énergie électrique produite par la chp	212.88 MWh	208.07 MWh
Énergie électrique exportée sur le réseau	187.97 MWh	181.24 MWh
Énergie électrique importée sur le réseau	17.80 MWh	15.88 MWh
Consommation de gaz naturel	80.79 T	76.03 T
Consommation de $CO2$	27.22 T	24.12 T
Consommation de $CO2$ de la source	4.97 T	10.21 T
Consommation de $CO2$ de la chaudière	22.25 T	13.91 T

Fig.4.33 présente le bilan économique des simulations pour les deux régulateurs de l'étude. De plus, la Tab.4.11 expose les valeurs numériques. En premier lieu, le vecteur électrique présente la particularité d'un poste de gain et de dépense pour le maraîcher. La vente est de 8681.1€ pour le régulateur classique et de 9079.6€ pour le régulateur avancé. La dépense

électrique représente 761.6€ pour le régulateur classique et 626.7€ pour le régulateur avancé. Nous pouvons observer que le régulateur avancé occasionne une augmentation des gains de la vente électrique et une baisse du coût d'achat électrique. En outre, il a été observé que la production électrique est réduite par l'unité de co-génération avec le régulateur avancé (Tab.4.11). Cette observation est générée par le fonctionnement de l'unité de co-génération étalée sur l'année de production par le régulateur avancé (Fig.4.31.a). En second lieu, l'achat de gaz naturel (Fig.4.33 et Tab.4.11) le régulateur avancé entraîne une baisse du coût par rapport au régulateur classique avec 626.7€ et 761.6€. Cet élément est conforme avec la baisse de la consommation de gaz naturel observée (Tab.4.10). Ensuite, l'achat de CO_2 par le régulateur avancé occasionne un doublement du coût par rapport au régulateur classique (×2.05). Cette augmentation du coût est occasionnée par une augmentation de l'apport de CO_2 par la source avec le régulateur avancé par rapport classique (Tab.4.10). Finalement, le bilan présente un gain de 5558.3€ par le régulateur classique et de 5400.3€ pour le régulateur avancé. Le bilan est avantageux pour le régulateur classique, en effet, l'augmentation du bilan du régulateur avancé est occasionnée par l'augmentation du coût CO_2 par un accroissement de l'utilisation de la source de CO_2 (Tab.4.10). Finalement, il est ajouté sur la Fig.4.33 le bilan économique avec le tarif de rachat actuel et le plus avantageux pour comparaison, la vente électrique réalisé est supérieure à 26000 €.

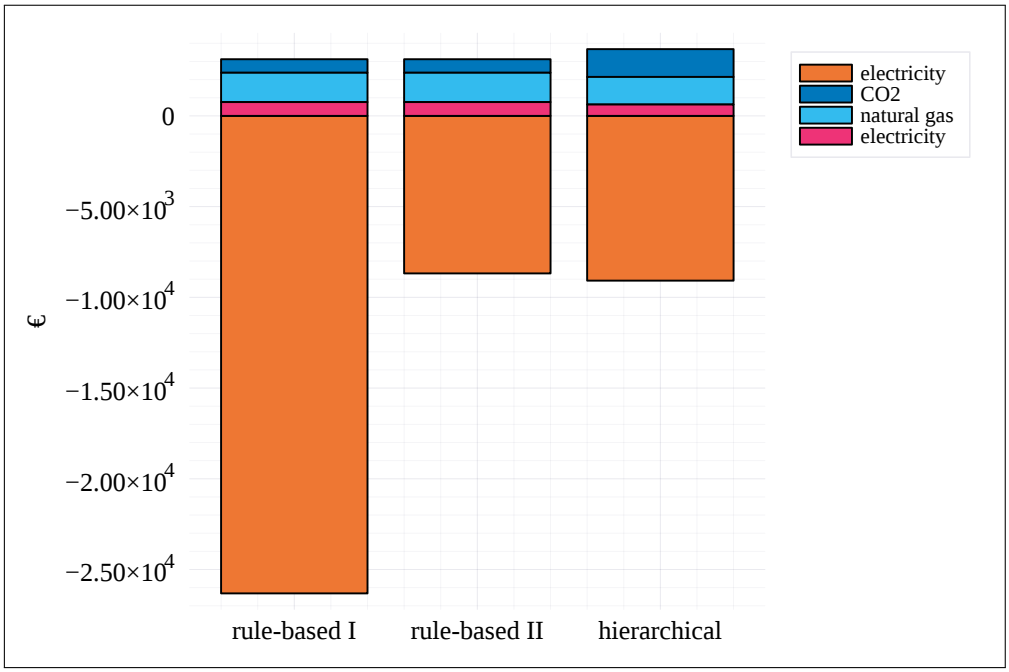

FIGURE 4.33 – Bilan économique des simulations sur l'année 2019.

Table 4.11 – Bilan économique des simulations sur l'année 2019 pour les deux régulateurs de l'étude (le signe négatif précise un gain et le signe positif un coût).

Controleur	Règles expertes (€)	Hiérarchique (€)
Vente électrique	−8681.1	−9079.6
Achat électrique	761.6	626.7
Achat gaz naturel	1615.7	1520.6
Achat CO_2	745.5	1531.9
Bilan	−5558.3	−5400.3

4.6 Discussion

Ce chapitre a présenté le pilotage d'un système multi-énergie. Ce système est constitué d'une serre agricole et des équipements supports à celle-ci. Deux régulateurs ont été développés, le premier est tiré de règles expertes et le second est un régulateur avancé à deux niveaux hiérarchiques de commande. Des simulations numériques ont été réalisées pour comparer le fonctionnement des équipements avec ces deux régulateurs. Le régulateur à base de règles expertes existe grâce à un tarif de rachat électrique avantageux pour le maraîcher. Il est prévu que ce contrat se termine et le régulateur hiérarchique est proposé en remplacement. La première information tirée des simulations est la possibilité de conserver l'unité de co-génération et de valoriser sa production énergétique. La production électrique de l'unité de co-génération est consommée par les demandes de la serre agricole ou exportée sur le réseau électrique.

D'un point de vue énergétique, il est observé une baisse de la consommation de chaleur pour le régulateur avancé par rapport au régulateur classique. Cette baisse est intéressante, car elle permet une réduction de la production de chaleur par la chaudière et l'unité de co-génération qui se répercute sur une baisse de la consommation de gaz naturel. Le régulateur avancé permet une meilleure allocation des ressources pour le vecteur de chaleur par rapport au régulateur basé sur des règles expertes. L'étude énergétique sur le vecteur électrique a mis en évidence une baisse de l'importation et de l'exportation électrique par le régulateur avancé par rapport au régulateur à base de règles expertes. Alors que par la même occasion, le profit résultant de la vente électrique est augmenté par le régulateur avancé. Ce régulateur réalise une meilleure allocation de fonctionnement sur l'année par l'unité de co-génération pour augmenter les profits avec une exportation électrique moindre. Néanmoins, il est à noter que l'amélioration des performances du régulateur avancé entraîne le besoin de modéliser le système ainsi que la prédiction des demandes énergétiques de la serre. La modélisation est rendue accessible par la méthodologie proposée dans ce travail de thèse. Concernant les prédictions des demandes

énergétiques, elles ont été considérées parfaites durant la simulation. En fonctionnement réel ces prédictions sont imparfaites. Une poursuite de l'étude semble nécessaire pour réaliser une étude de sensibilité avec des prédictions imparfaites et évaluer des méthodes pour réaliser ces prédictions.

Concernant le vecteur fertilisant, la consommation de CO_2 est plus importante avec le régulateur classique qu'avec le régulateur avancé. De plus, le régulateur avancé importe plus de CO_2 par la source. Il avait été considéré comme hypothèse que le fonctionnement de la chaudière à basse puissance permettait d'assouvir tous les besoins de CO_2 de la serre. L'hypothèse formulée par une utilisation accrue de la chaudière à basse puissance entraîne une baisse de la concentration de CO_2 et lorsqu'elle s'arrête, la source est sollicitée pour respecter la consigne de CO_2. Ceci entraîne une forte demande de CO_2 de la source. Ce point pourra être investigué et amélioré pour une poursuite de l'étude.

Nous pouvons maintenant faire le bilan des productions énergétiques du système de la serre agricole pour les deux simulations de l'étude. Les différents éléments observés sont :

- Une baisse de la production de chaleur par la chaudière est observée sur l'année de production pour le pilotage avancé par rapport au pilotage classique ;
- Une production de chaleur par l'unité de co-génération équivalente pour les deux régulateurs est observée ;
- Une production électrique équivalente pour les deux régulateurs avec un fonctionnement de l'unité de co-génération dispersée sur l'année pour le pilotage avancé ;
- Une baisse de la consommation de gaz naturel est réalisé avec le pilotage avancé ;
- Une augmentation de l'appoint en CO_2 par la source est obtenue par le pilotage avancé ;

D'un point de vue économique, le régulateur avancé entraîne une baisse du coût pour l'importation d'électricité, de gaz naturel et améliore le profit de vente électrique par rapport au régulateur classique. Néanmoins, le bilan final est à l'avantage du régulateur classique, car celui-ci requiert une moindre utilisation de la source de CO_2 que le régulateur avancé.

Des perspectives de poursuite du travail peuvent être esquissées en plus des travaux sur le vecteur CO_2 et la prédiction des demandes énergétiques. Tout d'abord, il peut être envisagé de réaliser une étude de sensibilité sur les paramètres de la fonction de coût de l'optimisation économique. Les deux simulations présentées ont utilisé les données météorologiques et économiques de l'année 2019. Il est intéressant de réaliser des simulations sur des années plus complètes et avec des tarifs énergétiques différents. Ensuite, des modifications du système énergétique de la serre agricole peuvent être esquissées pour une réduction du bilan énergétique et économique. Tout d'abord, le circuit de chaleur basse température ne possède pas de stockage. Il

a été observé que le fonctionnement de la chaudière pour injecter du CO_2 occasionne de temps en temps un apport de chaleur optionnel pour la serre agricole. Le stockage de chaleur sur le réseau basse température permettrait de stocker cette chaleur et de l'injecter ultérieurement lors d'un besoin. La seconde amélioration physique est la valorisation de la production de CO_2 par l'unité de co-génération afin de réduire l'appoint de CO_2 par la source. Actuellement, seul l'apport de CO_2 par la chaudière est valorisée.

4.7 Conclusion

Ce chapitre a exposé le pilotage d'un système multi-énergie avec une serre agricole et des équipements supports de production énergétique. Tout d'abord, la serriculture fut présentée puis un état de l'art a permis de mettre en lumière des méthodes de pilotage du système serre agricole. En second lieu, la serre de l'étude fut décrite. Ensuite, la méthodologie pour développer le pilotage du système multi-énergie est présentée, avec la modélisation de la serre agricole, le pilotage des variables climatiques, puis la présentation de deux régulateurs pour piloter les équipements supports à la serre agricole. Le premier régulateur est basé sur la méthode conventionnelle des serristes et le second régulateur a pour but de minimiser les coûts et il constitue l'apport de ce travail. Finalement, les résultats des simulations sont présentés et une discussion est exposée. Les résultats ont mis en évidence ; une baisse de la consommation de gaz naturel, une réduction du fonctionnement de la chaudière, une production de chaleur et électrique de l'unité de co-génération équivalente, un gain de vente électrique plus importante du régulateur avancé par rapport au régulateur basé sur des règles expertes. Néanmoins, la consommation de CO_2 par la source fut plus importante pour le régulateur avancé par rapport au régulateur classique, ce qui engendre un bilan économique du régulateur classique plus favorable sur l'année de simulation.

Quatrième partie

Conclusion et perspectives

Chapitre 5

Conclusion

Le pilotage d'un système multi-énergie offre des perspectives intéressantes pour augmenter l'efficacité, la flexibilité et le déploiement optimal de sources énergétiques plurielles et faisant intervenir des vecteurs énergétiques de natures diverses. Ce travail de recherche contribue à la méthodologie de conception des systèmes de pilotage énergétiques. Il s'est ainsi, tour à tour, intéressé à (1) formaliser la notion de systèmes multi-énergies et (2) investiguer les possibilités offertes par la commande MPC. Le point (1) a été approfondi dans [83]. Il a été utilisé dans le chapitre « serre » dont le système multi-énergie sous-jacent a été mis en exergue, et articulé avec le modèle dynamique global de la serre. L'étude du point (2) a comporté plusieurs étapes, précisant la voie proposée pour la recherche du modèle conceptuel de commande. Le modèle interne proposé pour être utilisé par la MPC à des fins de prédiction articule modélisation multi-physique et modélisation neuronale. La pertinence de la démarche a été mise en évidence dans un cadre plus général que celui des systèmes multi-énergie. In fine, elle a été mise en œuvre dans le cadre du pilotage d'une serre agricole, application cible de la thèse. Pour cela, un modèle multi-physique d'une serre a été développé à l'aide de Modelica, et le système multi-énergie support de la serre a été mis en évidence. Une architecture hiérarchique à deux niveaux a été proposée pour le pilotage du système multi-énergie. Les résultats obtenus ont été analysés en détail et comparés à ceux obtenus par un pilotage à base de règles expertes, mimant le pilotage réalisé par les serristes.

Le chapitre 1 a présenté en premier lieu la problématique de recherche. Il a été observé que l'énergie est fondamentale pour le fonctionnement de la société moderne, mais que des externalités sont occasionnées. Les systèmes multi-énergies peuvent répondre à réduire ces externalités pour l'augmentation de l'efficacité, de la flexibilité et du déploiement optimal. Néanmoins, il a été aussi vu que le pilotage de systèmes contenant plusieurs vecteurs énergétiques contient des défis pour la commande. Ainsi, un état de l'art des méthodes de pilotage de système a été présenté. Toutefois, la méthode avancée de pilotage exposé nécessite une modélisation de système dynamique, l'état de l'art a ensuite consisté à présenter les différentes catégories de modèle dynamique. L'état de l'art a pointé que l'articulation de ce travail sur

trois thématiques de recherche est un élément de différenciation fort par rapport aux travaux antérieurs. De plus, il a été aussi observé le peu d'études de système multi-énergies et de leur pilotage dans le domaine agricole.

Le chapitre 2 a présenté en détail le développement et la synthèse d'un régulateur avancé pour piloter un système dynamique. Cette présentation repose sur une méthodologie dont la synthèse est exposée à la section 2.1. Les différentes étapes de cette méthodologie sont : modélisation fine multiphysique, la génération et récolte des données, la paramétrisation du modèle boite noire, la synthèse du régulateur et l'évaluation du pilotage.

Le chapitre 3 a présenté la validation du développement et la synthèse d'un régulateur sur le *benchmark* d'automatique. Ce système dynamique est le quatre réservoirs. Il offre des caractéristiques intéressantes pour l'application de la méthodologie. Tout d'abord, il présente des non-linéarités, une simplicité de mise en oeuvre et des contraintes sur les hauteurs d'eau et les actionneurs sont présents. L'application de la méthodologie nous a renseigné que celle-ci est convenablement mise en oeuvre pour la synthèse du régulateur avancé basé sur la commande MPC neuronale. Par ailleurs, il a été évalué trois architectures de réseau de neurones, FNN, ResNet et PolyNet. Il est observé que les réseaux ResNet et PolyNet qui sont dérivés de méthode d'intégration numérique permettent d'améliorer la prédiction des états du système par rapport à Fnn.

Le chapitre 4 a exposé le pilotage du système multi-énergies de la serre agricole. Le pilotage des variables climatiques est réalisé par des régulateurs classiques (PID ou règles expertes) puis le pilotage des équipements énergétiques supports fut étudié. Dans un premier temps un régulateur extrait des méthodes des serristes fut appliqué. Ce régulateur est axé sur le fonctionnement de l'unité de co-génération sur des journées fixées dans un contrat de rachat fixe. Ce contrat arrive à son terme et il est proposé une commande hiérarchique contenant deux niveaux. Le premier niveau un régulateur optimisant les coûts économiques du système. La modélisation par le concept du hub énergétique est réalisée avec le bilan de puissance. Le second régulateur contient une commande MPC. Ce régulateur a été synthétisé à l'aide de la méthodologie développée dans le chapitre 2 et 3. Cette commande hiérarchique a été comparée avec le régulateur de règles expertes. Il a été observé une réduction de la production de chaleur de la chaudière, une augmentation de l'apport de CO_2 par le réseau et une baisse de la consommation de gaz naturel.

Chapitre 6

Perspectives

Certaines limites du travail réalisé permettent d'esquisser des perspectives de recherche. Les perspectives présentées sont aussi bien exploratoires qu'applicatives, elles concourent toutes à l'amélioration de la méthodologie développée dans ce travail.

6.1 Perspectives technologiques

Amélioration de l'implémentation numérique L'amélioration de la méthodologie a pour but d'améliorer l'implantation de celle-ci pour atteindre un démonstrateur. Les démonstrateurs permettent de présenter aux partenaires du CEA les développements technologiques. Actuellement, certaines barrières logicielles entravent une bonne mise en pratique d'un démonstrateur pour exposer les capacités de pilotage des systèmes multi-énergies. L'amélioration de la méthodologie et de son implémentation permettrait d'atteindre un démonstrateur numérique. Les points suivants identifiés nécessitent une amélioration :

- Améliorer l'interface de Modelica avec Julia, l'utilisation de FMU et l'interface avec Simulink impose des contraintes fortes et ne permet pas une réalisation idéale et robuste du simulateur numérique. Une solution envisageable est par exemple, l'appel de la fonction Julia par Modelica.
- Réaliser la compilation et la simulation du programme Modelica sur des serveurs de calcul sous Linux. Actuellement, le compilateur Dymola restreint la réutilisation du programme compilé Modelica pour l'exécuter sur une machine ne possédant pas Dymola. Par ailleurs, un programme Modelica créé avec Dymola est mystérieusement non compilable avec un autre compilateur, alors que Modelica possède une spécification réalisée par l'association Modelica [240].

Application de la méthodologie sur un système multi-énergie physique Une première perspective de recherche est l'application de la méthodologie pour développer un régulateur afin de piloter un processus réel. En effet, durant la thèse de recherche la méthodologie a été

appliquée sur un système multi-énergie numérique.

Agrandissement du cadre de systèmes multi-énergies Nous avons présenté dans ce manuscrit le cadre de modélisation de systèmes multi-énergies par le hub énergétique, chapitre 1. Cette structure a pour but la modélisation des systèmes multi-énergies avec la prise en compte des différents vecteurs énergétiques. La serre agricole étudiée dans ce travail contient des vecteurs énergétiques (chaleur, électricité, gaz naturel) et aussi un gaz fertilisant avec le dioxyde de carbone. Nous pouvons aussi ajouter l'eau ou le rayonnement pour la plante. Une perspective de recherche est la prise en compte globale des flux pour le système et pas seulement les vecteurs énergétiques afin de les piloter conjointement et obtenir des synergies.

6.2 Perspectives sur la commande

Planification La présentation de la commande hiérarchique du chapitre 4 avec deux étages, a entraîné un premier niveau contenant un optimiseur pour calculer les points de consignes. Puis le second niveau qui implémente la commande MPC pour piloter les actionneurs. Dans ce travail, le bilan de puissance avec les équations algébriques simples du système multi-énergie a été utilisé pour modéliser un problème d'optimisation. Ce bilan de puissance entraîne la reformulation des équations pour chaque système multi-énergie. Une perspective de recherche est d'implémenter un étage de calcul avec l'apprentissage par renforcement pour obtenir les consignes minimisant les coûts économiques. Ce régulateur peut s'appuyer sur les avancées scientifiques réalisées dans le domaine [241].

Commande prédictive économique Dans ce travail de recherche, la commande utilisée pour piloter la serre agricole est une commande hiérarchique avec un régulateur planificateur puis un régulateur basé sur la commande MPC. Il existe une commande prédictive économique avec un coût économique du processus à la place d'une fonction de coût quadratique. Cette commande permet de remplacer la commande hiérarchique à deux niveaux par un seul niveau [242].

Commande prédictive adaptative La commande prédictive présentée dans ce travail de thèse utilise un modèle neuronal dont l'apprentissage a été réalisé avant la synthèse du régulateur à l'aide des données générées par le simulateur. En acceptant une dégradation des performances au démarrage du pilotage, il est envisageable d'obtenir une commande prédictive

adaptative dont l'apprentissage du modèle est réalisé en même temps que le pilotage. Ce fonctionnement tirerait parti de la capacité d'extrapolation du ResNet observé dans ce travail de recherche. De plus, une MPC adaptative a été explorée dans des travaux antérieurs, par exemple dans [243] ou [244]. Néanmoins, ces travaux n'ont pas utilisé de ResNet pour la modélisation, ainsi la comparaison des performances est envisageable.

Commande prédictive robuste La commande prédictive présentée dans ce travail de thèse ne tient pas compte des perturbations induites par des capteurs de mesure ou l'erreur de modélisation. Une perspective de travail est d'appliquer des techniques et méthodes déjà existantes dans la littérature, tel que min-max MPC [245], multistage MPC [246], ou la MPC en tube [247] tout en conservant la modélisation réalisée par un réseau ResNet à l'aide de données.

6.3 Perspectives sur la prédiction du système dynamique

Comparaison des architectures neuronales Les évaluations des architectures réalisées dans ce travail de recherche ont porté sur le Fnn, le ResNet et le PolyNet. Il peut être intéressant de continuer la comparaison des architectures de réseau de neurones pour l'identification de système dynamique. Un nombre conséquent d'architectures de neurones artificiels sont proposées dans la littérature pour la classification des images. Il serait intéressant d'évaluer l'applicabilité pour l'identification de système dynamique et de pouvoir les comparer. En effet, pour une performance équivalente il existe peut-être une architecture qui permet une réduction du nombre de paramètres ou une réduction du temps d'apprentissage par rapport au ResNet. Par exemple, avec le réseau DenseNet [248], il est observé que pour une performance équivalente au ResNet, le réseau permet une réduction du nombre de paramètres.

Réseaux de neurones convexes L'utilisation (comme modèle interne) des réseaux de neurones pour prédire l'état d'un système dynamique engendre une MPC et un problème d'optimisation associé non-linéaires. Le caractère non-linéaire est dû à l'utilisation de fonctions d'activation non-linéaires. Ainsi, le minimum global est obtenu seulement en évaluant tous les minima locaux, ce qui dans la pratique est impossible. Une perspective de recherche consiste à reformuler le problème d'optimisation non convexe pour un problème d'optimisation convexe avec des contraintes coniques. Un réseau de neurone convexe appelé *Input Convex Neural Networks* a été proposé dans [249]. Il serait intéressant de comparer les performances par rapport à ceux obtenus dans ce travail de thèse avec le ResNet (temps d'apprentissage, généralisation,

...). En outre, il a été utilisé dans [250] avec une commande MPC. Cette voie est en lien avec la branche scientifique : *Disciplined Convex Programming (DCP)* [251]. Il existe des outils de reformulation numérique tels que Convexe.jl dans le langage Julia [252]. En outre, il existe le solveur de résolution Cosmo [170] adapté à ce problème d'optimisation.

Réseaux de neurones frugaux L'élaboration et la mise en œuvre d'une commande NN-MPC est énergivore pour l'apprentissage des réseaux d'une part et pour optimisation non-linéaire réalisée à chaque pas d'échantillonnage d'autre part. La réflexion sur ce point pourrait être approfondie, en analysant, à performance comparable, les coûts énergétiques associés à l'utilisation de MPC linéaire ou non linéaire, convexe ou non. Afin de réduire la demande énergétique pour réaliser l'apprentissage des réseaux, trois perspectives sont envisageables. La première est de sélectionner rigoureusement l'architecture. Nous avons pu observer que PolyNet nécessite plus de temps d'apprentissage que le réseau ResNet. La perspective de comparaison des architectures pour réduire le nombre de paramètres ouvre la possibilité de la réduction du temps d'apprentissage. La seconde perspective est de réaliser l'optimisation des hyperparamètres avec deux objectifs. Le premier objectif est la réduction de la mesure de la fidélité et le second objectif est le nombre de paramètres du réseau ou d'autre méthode de mise à l'échelle du réseau de neurones [253]. Finalement, l'algorithme de rétropropagation du gradient peut aussi être questionné, car celui-ci est itératif. La méthode de l'*Extreme learning machine* pour la paramétrisation du réseau Fnn par un algorithme non itératif mérite d'être considérée [254].

De nombreuses perspectives de recherche sont esquissées dans les sections précédentes. Ces perspectives sont présentes sur les trois thématiques transverses, le domaine énergétique et les systèmes multi-énergies, l'algorithme de commande et la modélisation par des réseaux de neurones. Les aspects ayant trait à l'amélioration de l'implémentation numérique nous semblent prioritaires pour atteindre un démonstrateur numérique de pilotage d'un système multi-énergies.

Cinquième partie

Annexes

Annexe A

Liste des figures

LISTE DES FIGURES

1.1 Exemple d'une unité de co-génération. 16
1.2 Exemple d'une unité de tri-génération. 16
1.3 Exemple d'une unité multi-entrées et multi-sorties. 16
1.4 Schéma d'une boucle fermée. 18
1.5 Schématisation d'une commande hiérarchique, adapté de [34]. 20
1.6 Modèle boîte noire d'un système dynamique. 24
1.7 Exemple d'un hub énergétique, tiré de [79] ©[2011] IEEE. 25
1.8 Plan de la thèse de recherche. 28

2.1 Schématisation de la méthodologie proposée. 33
2.2 Schématisation de la simplification du neurone biologique, extrait de [108]. . . 37
2.3 Schématisation du neurone artificiel, adapté de [112] avec à gauche la représentation produit par élément et à droite la représentation vectorielle. 38
2.4 Schématisation du perceptron multi-couches profond, avec une entrée, une sortie et quatre couches cachées. 39
2.5 Schématisation d'une unité de ResNet [119]. 40
2.6 Schématisation du ResNet avec deux couches. 41
2.7 Schématisation d'une unité de PolyNet [126]. 42
2.8 Schématisation du PolyNet avec deux couches. 43
2.9 Schématisation du Resnet (haut), PolyNet (milieu) et Fnn (bas) avec les couches d'entrée et de sortie. (neurones en gris). 45
2.10 Algorithme optimisation des hyperparamètres du réseau de neurones (prédicteur des états). 46
2.11 Schématisation d'une boucle de contrôle avec la MPC. 48
2.12 Horizon glissant occasionné par la résolution itérative de la commande MPC. . 50
2.13 Algorithme de la commande MPC. 51

3.1 Schéma du système quatre réservoirs. 58
3.2 Modélisation fine du système 4 réservoirs avec Dymola. 62

3.3 Portion de données générées par les simulations Dymola. (a) Présente les signaux des entrées de commande avec des séquences binaireq pseudo-aléatoires. (b) Présente les hauteurs d'eau des réservoirs associées. (c) Présente les signaux des entrées de commande avec des échelons aléatoires. (d) Présente les hauteurs d'eau des réservoirs associées. 64
3.4 Séparation des données selon la hauteur d'eau des réservoirs 1 et 2. (a) Présente les hauteurs d'eau de h_1, h_2 entre 0.4m to 0.8m. Ces données constituent les données d'entrainement et de validation. (b) Présente les hauteurs d'eau de h_3, h_4 avec les pairs de h_1, h_2 à partir des données de (a). Ces données constituent les données d'entrainement et de validation. (c) Présente les hauteurs d'eau de h_1, h_2 lorsque les hauteurs d'eau sont en dehors de 0.4m et 0.8m. Ces données constituent les données de test. (d) Présente les hauteurs d'eau de h_3, h_4 associées aux pairs de h_1, h_2. Ces données constituent les données de test. 65
3.5 Procédure de pré-traitement des données à partir des données brutes pour obtenir les données d'entrainement, de validation et de test. 66
3.6 Fonction d'activation, (a) Softsign : $\frac{x}{1+|x|}$, (b) Swish : $\frac{x}{1+e^{-x}}$, (c) Tanh : $tanh(x)$ et (d) Sigmoïde $\frac{1}{1+e^{-x}}$. 68
3.7 boîte à moustache de la fonction de coût selon les réseaux Fnn, ResNet et PolyNet, (a) avec les données d'entrainement, (b) avec les données de validation et (c) avec les données de test. 69
3.8 Boxplot de fonctions de coût en fonction des hyperparamètres des réseaux selon, (a, f, k) le nombre de couches cachées, (b, g, i) le nombre de neurones, (c, h, m) la fonction d'activation, (d, i ,n) l'optimiseur et (e, j, o) le nombre d'époque. 72
3.9 Boîte à moutache avec la fonction de coût selon le nombre de couches des réseaux (a) ResNet et (b) PolyNet avec les données de test. 73
3.10 boîte à moutache avec la fonction de coût selon le temps d'apprentissage des réseaux (a) Fnn, (b) ResNet et (c) PolyNet. 74
3.11 Algorithme d'optimisation des paramètres de la MPC. 80
3.12 Schématisation des régulateurs à base de MPC synthétisées pour piloter le système quatre réservoirs. 81
3.13 Environnement numérique de simulation contenant la FMU et le régulateur MPC à base de neurones artificiels. 83
3.14 boîte à moustache de l'IAE pour les 150 pilotages du système quatre réservoirs avec la MPC basée sur le modèle de prédiction Fnn. 83

3.15 IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture Fnn. 85
3.16 Trajectoires de pilotage du système quatres réservoirs avec l'architecture Fnn. Les carrés jaune indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4. 86
3.17 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture Fnn : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2. 87
3.18 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture Fnn : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4. 88
3.19 Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture Fnn : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b. 88
3.20 boîte à moustache des 150 pilotages du système quatre réservoirs avec l'architecture ResNet et les paramètres de la MPC. L'IAE est présentée en fontion des états du système. 89
3.21 IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture ResNet. 90
3.22 Trajectoires de pilotage du système quatres réservoirs avec l'architecture ResNet. Les carrés jaunes indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4. 91
3.23 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture ResNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2. 92
3.24 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture ResNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4. 93

3.25 Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture ResNet : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b. 93
3.26 boîte à moustache des 150 pilotages du système quatre réservoirs avec l'architecture PolyNet. L'IAE est présentée en fontion des états du système. 94
3.27 IAE moyen sur les quatres états en fonction des paramètres de la MPC a) κ_Q, b) κ_R et c) l'horizon N avec l'architecture PolyNet. 95
3.28 Trajectoires de pilotage du système quatres réservoirs avec l'architecture PolyNet. Les carrés jaunes indiquent les points de consigne. (a) Présente les 151 trajectoires avec les hauteurs d'eau h_1 et h_2. (b) Présente les 151 trajectoires avec les hauteurs d'eau h_3 et h_4. (c) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_1 et h_2. (d) Présente la trajectoire ayant entrainée la plus faible IAE selon les hauteurs d'eau h_3 et h_4. 96
3.29 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture PolyNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir un, h_1. (b) Présente la hauteur d'eau du réservoir deux, h_2. 97
3.30 Résultats de la simulation ayant entraînée la plus faible IAE avec l'architecture PolyNet : hauteur d'eau en fonction du temps. (a) Présente la hauteur d'eau du réservoir trois, h_3. (b) Présente la hauteur d'eau du réservoir quatre, h_4. 98
3.31 Résultats de la simulation ayant entrainée la plus faible IAE avec l'architecture PolyNet : débit des pompes en fonction du temps. (a) Présente le débit de la pompe q_a. (b) Présente le débit de la pompe q_b. 98

4.1 Surface mondiale de serre agricole [203], Rabobank. 104
4.2 Serre agricole légère. 105
4.3 Serre agricole lourde [205], Gestebier, CC BY-SA 4.0, via Wikimedia Commons. 106
4.4 Équipements et flux pour gérer les variables climatiques [207]. 108
4.5 Commande hierarchique de la serre agricole de [216] avec les deux niveaux de commande, ©[2003] IEEE. 110
4.6 Illustration des bilans des variables climatiques avec la serre agricole et des équipements supports. 116
4.7 Schéma Modelica du processus serre agricole avec le système multi-énergies et la serre. 118
4.8 Schéma Modelica de la serre agricole tiré de [92] et adapté pour le cas d'étude. 118

4.9 Schéma de la serre agricole avec les variables climatiques, les équipements supports et les régulateurs. 120
4.10 Schéma de la serre agricole avec le régulateur classique. 123
4.11 Schéma de principe du régulateur à base de règles expertes et les variables à dispositions. 124
4.12 Schéma de la commande avancée développée pour le système multi-énergies de la serre agricole. 125
4.13 Schéma de la serre agricole avec le régulateur avancé. 126
4.14 Schéma de principe du régulateur avancé et les variables à dispositions. 127
4.15 Hub énergétique schématisé avec le système multi-énergies support de la serre agricole. 128
4.16 Réseau de neurones utilisé pour la prédiction des états du système multi-énergie. 140
4.17 Variables météorologiques de l'année 2019 avec (a) la température extérieure, (b) la température du ciel, (c) le rayonnement global, (d) la vitesse du vent et (e) l'humidité relative. 142
4.18 Environnement numérique de simulation contenant la FMU et un régulateur. . 143
4.19 Coût de l'électricité un jour à l'avance sur l'année 2019. 146
4.20 Monotone des coûts électriques sur l'année 2019. 146
4.21 boîte à moustache de la fonction de coût selon les temps d'échantillonnage des données. (a) Présente les boîtes à moustache avec les données de validation. (b) Présente les boîtes à moustache avec les données de test. 149
4.22 boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $10s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur. 150
4.23 boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $30s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur. 151
4.24 boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $60s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur. 152

4.25 boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $120s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur. 153
4.26 boîte à moustache de la fonction de coût avec les données de validation échantillonnées à $300s$ selon les hyperparamètres : (a) en fonction du nombre de couches cachées, (b) en fonction du nombre de neurones, (c) en fonction du nombre d'epochs, (d) en fonction de l'optimiseur. 154
4.27 Pilotage avec le régulateur classique, (a) monotones de puissance thermique, (b) la température de la serre, (c) l'état de charge du stockage, (d) le débit de CO_2 de la source, (e) le débit de CO_2 de la chaudière, (f) la commande de l'unité de co-génération, (g) la commande de la chaudière, (h) la température extérieure, (i) le rayonnement. 158
4.28 Pilotage avec le régulateur classique, (a) monotones de puissance électrique, (b) le coût de l'électricité, (c) le rayonnement. 161
4.29 Série temporelle du rayonnement solaire et de la puissance électrique. 163
4.30 Pilotage avec le régulateur avancé, (a) monotones de puissance thermique, (b) la température de la serre, (c) l'état de charge du stockage, (d) le débit de CO_2 de la source, (e) le débit de CO_2 de la chaudière, (f) la commande de l'unité de co-génération, (g) la commande de la chaudière, (h) la température extérieure, (i) le rayonnement. 164
4.31 Pilotage avec le régulateur avancé, (a) monotones de puissance électrique, (b) le coût de l'électricité, (c) le rayonnement. 166
4.32 Pilotage avec le régulateur classique et avancé, (a) monotone du coût électrique, (b) la commande par le régulateur classique de l'unité de co-génération, (c) la commande par le régulateur avancé de l'unité de co-génération, (d) la demande électrique, (e) la demande de chaleur de la serre avec le régulateur classique, (f) la demande de chaleur de la serre avec le régulateur avancé. 169
4.33 Bilan économique des simulations sur l'année 2019. 171

E.1 Schématisation d'un neurone RNN. 200
E.2 Schématisation de DenseNet [248]. 204

F.1 Illustration de H [271]. 211
F.2 Modèle multinœud d'un stockage thermique [273]. 215

F.3 Trois étages pour la modélisation de la chaudière [230]. 216

Annexe B

Liste des tableaux

LISTE DES TABLEAUX

3.1 Paramètres du système quatre réservoirs [187]. 60
3.2 Paramètres des séquences. 63
3.3 Hyperparamètres acceptables du réseau de neurones. 68
3.4 MSE loss with train data. 70
3.5 MSE loss with validation data. 70
3.6 MSE loss with test data. 70
3.7 Temps d'entrainement (s). 74
3.8 Résultats de l'optimisation des hyperparamètres avec l'architure Fnn ayant entrainée les 25 plus faibles MSE avec les données de test. 76
3.9 Résultats de l'optimisation des hyperparamètres avec l'architure ResNet ayant entrainée les 25 plus faibles MSE avec les données de test. 77
3.10 Résultats de l'optimisation des hyperparamètres avec l'architure PolyNet ayant entrainée les 25 plus faibles MSE avec les données de test. 78
3.11 Paramètres admissibles pour l'algorithme 3.11 82
3.12 Points de consigne en régime établi pour le pilotage du système quatre réservoirs avec la commande MPC. 84
3.13 Paramètres de la MPC et de l'IAE. 99

4.1 Paramètres synthétiques du système serre agricole. 113
4.2 Paramètres renseignés pour le problème d'optimisation. 145
4.3 Paramètres des coûts économiques renseignés pour le problème d'optimisation. 145
4.4 Paramètres des facteurs de conversion des équipements. 145
4.5 Hyperparamètres autorisés du réseau de neurones. 147
4.6 Paramètres de la MPC. 148
4.7 Fonction de coût avec les données de validation. 149
4.8 Fonction de coût avec les données de test. 150
4.9 Hyperparamètres minimisants la fonction de coût. 155
4.10 Bilan énergétique des simulations. 170

4.11 Bilan économique des simulations sur l'année 2019 pour les deux régulateurs de l'étude (le signe négatif précise un gain et le signe positif un coût). 172

Annexe C

Communications scientifiques

Ce travail de thèse a donné lieu aux publications suivantes :

C.1 Revues internationales

Blaud, P., C., Haurant, P., Claveau, F., Lacarrière, B., Chevrel, P., Mouraud, A., 'Modelling and control of multi-energy systems through Multi-Prosumer Node and Economic Model Predictive Control', International Journal of Electrical Power & Energy Systems, 2020, 118, doi : 10.1016/j.ijepes.2019.105778.

Blaud, P., C., Chevrel, P., Claveau, F., Haurant, P., Mouraud, A.,'From multi-physics models to neural network for predictive control synthesis', Optimal Control Applications and Methods, 2021, doi : 10.1002/oca.2845.

Soumis

Blaud, P., C., Chevrel, P., Claveau, F.,Haurant, P., Mouraud, A., 'ResNet and PolyNet based identification and (MPC) control of dynamical systems : a promising way', 2022. (en attente de notification d'une deuxième lecture par les pairs)

Rédaction

Blaud, P., C., Chevrel, P., Claveau, F.,Haurant, P., Mouraud, A., 'Multi-flux optimisation of a greenhouse with a model predictive control'.

C.2 Conférences internationales

Blaud, P., C., Chevrel, P., Claveau, F.,Haurant, P., Mouraud, A., 'Four MPC implementations compared on the Quadruple Tank Process Benchmark : pros and cons of neural MPC', 18th

IFAC Workshop on Control Applications of Optimization, 2022, Gif-Sur-Yvette, France, hal :hal-03631171.

Ce travail de thèse a aussi donné lieu aux travaux suivants :

C.3 Présentations

— Journée des doctorants du LITEN, CEA Grenoble, 2019, 2020, 2021, présentation intitulée : Pilotage distribué de systèmes multi-énergies en réseau ;
— Nantes Machine Learning Meetup, 2021, présentation intitulée : 'Des modèles physiques aux réseaux de neurones pour la commande prédictive'.

C.4 Rapports

— Journée des doctorants du site nantais de l'école doctorale MathSTIC, 2021, rapport intitulé : Pilotage distribué de systèmes multi-énergies en réseau ;

C.5 Enseignements

— TP automatique numérique (régulateurs PID et RST) à l'ICAM de Nantes semestre automne 2019 et 2020 ;
— Cours et TP commande prédictive à l'IMT Atlantique semestre printemps 2020 et 2021 ;
— Cours et TP réseaux de neurones pour l'identification et la prédiction de systèmes dynamiques, semestre printemps 2021 ;

Annexe D

Mesure de la fidelité de prédiction pour la regression

D.1 Erreur quadratique moyenne

Erreur quadratique moyenne, *Mean Square Error* (MSE) est [135] :

$$g_{loss} = \frac{1}{N_b} \sum_{n=1}^{N_b} (\bar{x}_n - \hat{x}_n)^2 \tag{D.1}$$

avec N_b le nombre total d'échantillon, $\hat{x}_n$ la sortie du modèle g et $\bar{x}_n$ les échantillons cibles.

D.2 Racine de l'erreur quadratique moyenne

Racine de l'erreur quadratique moyenne, *Root Mean Square Error* (RMSE) est [255] :

$$g_{loss} = \sqrt{\frac{1}{N_b} \sum_{n=1}^{N_b} (\bar{x}_n - \hat{x}_n)^2} \tag{D.2}$$

avec N_b le nombre total d'échantillon, $\hat{x}_n$ la sortie du modèle g et $\bar{x}_n$ les échantillons cibles.

D.3 Erreur moyenne absolue

Erreur moyenne absolue, *Mean Absolute error* (MAE) est [255] :

$$g_{loss} = \frac{1}{N_b} \sum_{n=1}^{N_b} |\bar{x}_n - \hat{x}_n| \tag{D.3}$$

avec N_b le nombre total d'échantillon, $\hat{x}_n$ la sortie du modèle g et $\bar{x}_n$ les échantillons cibles.

D.4 Erreur absolue moyenne en pourcentage

Erreur absolue moyenne en pourcentage, *Mean Absolute Percentage Error* (MAPE) est [255] :

$$g_{loss} = \frac{1}{N_b} \sum_{n=1}^{N_b} \left| \frac{\bar{x}_n - \hat{x}_n}{\bar{x}_n} \right| \tag{D.4}$$

avec N_b le nombre total d'échantillon, $\hat{x}_n$ la sortie du modèle g et $\bar{x}_n$ les échantillons cibles.

Annexe E

Réseaux de neurones non considérés dans cette étude

Dans le cadre de ce projet de recherche, des réseaux de neurones ont été abordés pour l'étude, néanmoins ils n'ont pas été sélectionnés pour une implémentation numérique.

E.1 Réseaux récurrents

E.1.1 RNN

Le neurone avec récurrence est le plus simple de mise en oeuvre, il présente une modification d'un neurone artificiel avec un retour de la sortie précédente, il est appelé *Vanilla Recurrent Neural Network* Vanilla RNN [256]. RNN permet de prendre en compte les dynamiques temporelles entre les données [257] comparativement au neurone artificiel classique qui traite des données statiques. Une illustration du RNN est visible avec la Fig.E.1. La formulation mathématique de Vanilla RNN est :

$$h[k] = \sigma(Wv[k] + Uh[k-1] + b) \tag{E.1}$$

avec $h[k]$ la sortie de la cellule RNN à l'instant k, σ la fonction d'activation, W, U les matrices de poids, $h[k-1]$ la sortie de la cellule à l'instance $k-1$ et finalement b le seuil.

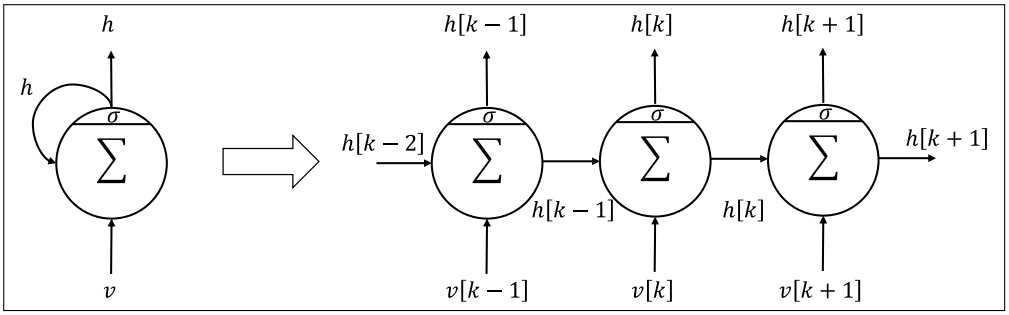

Figure E.1 – Schématisation d'un neurone RNN.

Il existe deux autres unités de récurrence appelées *Long short-term memory* (LSTM) et *Gated recurrent unit* (GRU). Le principe est équivalent au RNN. L'avantage de LSTM et GRU est la réduction de difficulté d'entrainement avec l'événement de la disparition du gradient. Les unités LSTM et GRU sont présentées dans les sections suivantes.

E.1.2 Long short-term memory (LSTM)

Long short-term memory (LSTM) est une unité avec mise en oeuvre de récurrence comme RNN. LSTM a été proposé en premier lieu en 1997 [258]. LSTM permet de remédier à la difficulté de la disparition du gradient pour mémoriser les entrées/sorties passées lors de l'entrainement du RNN [259]. LSTM est utilisé pour les applications de reconnaissance de l'écriture manuscrite à l'analyse audio ou vidéo [259] et aussi l'accélération de simulation de modèles biologiques contenant des équations aux dérivées partielles [260]. LSTM possède trois portes avec récurrence : la porte d'entrée, la porte d'oublie et la porte de sortie. La formulation mathématique du LSTM est égale à :

$$\tilde{c}[k] = tanh(W_c v[k] + U_c h[k-1] + b_c) \tag{E.2}$$

$$c[k] = o[k] \odot c[k-1] + e[k] \odot \tilde{c}[k] \tag{E.3}$$

$$h[k] = s[k] \odot tanh(c[k]) \tag{E.4}$$

avec $\tilde{c}[k]$ le vecteur de mémorisation candidat avec $tanh$ la fonction d'activation, $v[k]$ l'entrée de LSTM et U_c, W_c, b_c les matrices de poids et le seuil. $c[k]$ est le vecteur de mémorisation avec $o[k]$ la porte d'oubli et $e[k]$ la porte d'entrée. $h[k]$ est la sortie de l'unité LSTM avec $s[k]$ la porte de sortie. Par ailleurs, $\odot$ est le produit d'Hadamard. [1] En définitive, $h[k]$ est le vecteur de sortie contenant la porte d'oublie $o[k]$ et $tanh$ la fonction d'activation. Les portes de l'unité LSTM sont :

$$e[k] = \sigma(W_e v[k] + U_e h[k-1] + b_e) \tag{E.5}$$

$$o[k] = \sigma(W_o v[k] + U_o h[k-1] + b_o) \tag{E.6}$$

$$s[k] = \sigma(W_s v[k] + U_s h[k-1] + b_s) \tag{E.7}$$

avec $e[k]$, $o[k]$ et $s[k]$ les portes d'entrée, d'oublie et de sortie. W_e, U_e, W_o, U_o, W_s et U_s sont les matrices de poids et b_e, b_o et b_s sont les seuils des portes.

1. Produit matriciel d'Hadamard : $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \odot \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$

E.1.3 Gated recurrent unit (GRU)

GRU est un réseau avec récurrence et il possède deux portes (portes de mise à jour et porte de réinitialisation) [261]. Le GRU est dans la même tendance que LSTM avec une volonté de simplification [262]. La formulation mathématique du GRU est égale à :

$$h[k] = (1 - z[k]) \odot h[k-1] + z[k] \odot \tilde{h}[k] \tag{E.8}$$

$$\tilde{h}[k] = tanh(W_h v[k] + U_h(r[k] \odot h[k-1]) + b_h \tag{E.9}$$

avec $h[k]$ l'état caché et $\tilde{h}[k]$ l'état caché candidat. W_h, U_h sont les matrices de poids et b_h est le biais, $tanh$ est la fonction d'activation. $v[k]$ est l'entrée, $\odot$ est le produit d'Hadamard. $z[k]$ est la porte de mise à jour et $r[k]$ est la porte de réinitialisation, telle que :

$$z[k] = \sigma(W_z v[k] + U_z h[k-1] + b_z) \tag{E.10}$$

$$r[k] = \sigma(W_r v[k] + U_r h[k-1] + b_r) \tag{E.11}$$

avec $z[k]$ la porte de mise à jour et $r[k]$ la porte de réinitialisation, σ la fonction d'activation sigmoïde, W_z, W_r, U_z et U_r les matrices de poids, b_z et b_r les seuils, $v[k]$ l'entrée et $h[k-1]$ l'état caché à l'instant précédent.

E.2 Réseaux dérivés de méthode de résolution numérique

E.2.1 RKNN

Un réseau de neurones dérivé d'un ordre élevé de Runge-Kutta est présenté dans le document [263], nommé *Runge-Kutta Neural Networks* (RKNN). Les auteurs ont directement utilisé la relation Runge-Kutta d'ordre 4 pour réaliser le réseau :

$$y[k+1] = y[k] + \frac{1}{6}\Delta_k(\kappa_0 + 2\kappa_1 + 2\kappa_2 + \kappa_3) \tag{E.12}$$

Puis,

$$\kappa_0 = \mathcal{G}(y[k]) \tag{E.13}$$

$$\kappa_1 = \mathcal{G}(y[k] + \frac{1}{2}\Delta_t \kappa_0) \tag{E.14}$$

$$\kappa_2 = \mathcal{G}(y[k] + \frac{1}{2}\Delta_t \kappa_1) \tag{E.15}$$

$$\kappa_3 = \mathcal{G}(y[k] + \Delta_t \kappa_2) \tag{E.16}$$

avec $y[k+1]$ la sortie du réseau de neurones et la solution à $k+1$, $y[k]$ l'entrée du réseau, Δ_k la période entre deux échantillons. $\mathcal{G}$ est la transformation non linéaire du réseau de neurones. Les auteurs ont classé les réseaux en deux catégories : *direct-mapping neural network* (DMNN) et les réseaux qui apprennent le taux de changement des états du système dénommés *Runge-Kutta neural network* (RKNN). Par ailleurs, dans le document [264] il est indiqué que les réseaux AlexNet, VGGNET, GoogLeNet, ResNet originales sont des DMNN alors que les réseaux ResNet avec préactivation, DenseNets et CliqueNets sont des RKNN.

E.2.2 DenseNet

L'architecture DenseNet pour *Densely connected neural networks* [248] est dérivée du ResNet. L'objectif de DenseNet est la réduction du nombre de paramètre par rapport à ResNet, ainsi les auteurs précisent des performances équivalentes avec une réduction du nombre de paramètre. DenseNet modifie ResNet par l'entrée de la couche cachée par une concaténation de toutes les sorties des couches cachées précédentes, Fig.E.2. La formulation mathématique de DenseNet est :

$$h^l = \mathcal{G}(v^l) \tag{E.17}$$

avec h^l la sortie de la cellule DenseNet considérée, v^l l'entrée de la cellule considérée qui est égale à la concaténation de toutes les précédentes couches cachées du DenseNet : $v = cat(h^1, \ldots, h^{l-1})$. Ainsi il peut être facilement observé que le réseau DenseNet est applicable pour un nombre de couches cachées supérieures à une. Finalement, $\mathcal{G}$ est la transformation neuronale non linéaire constituée de neurones artificiels.

E.2.3 FractalNet

Fractal neural network (FractalNet) [265] est relié à la méthode d'intégration de Runge Kutta [125]. La formulation de Runge-Kutta d'ordre deux est équivalente à la méthode d'Euler

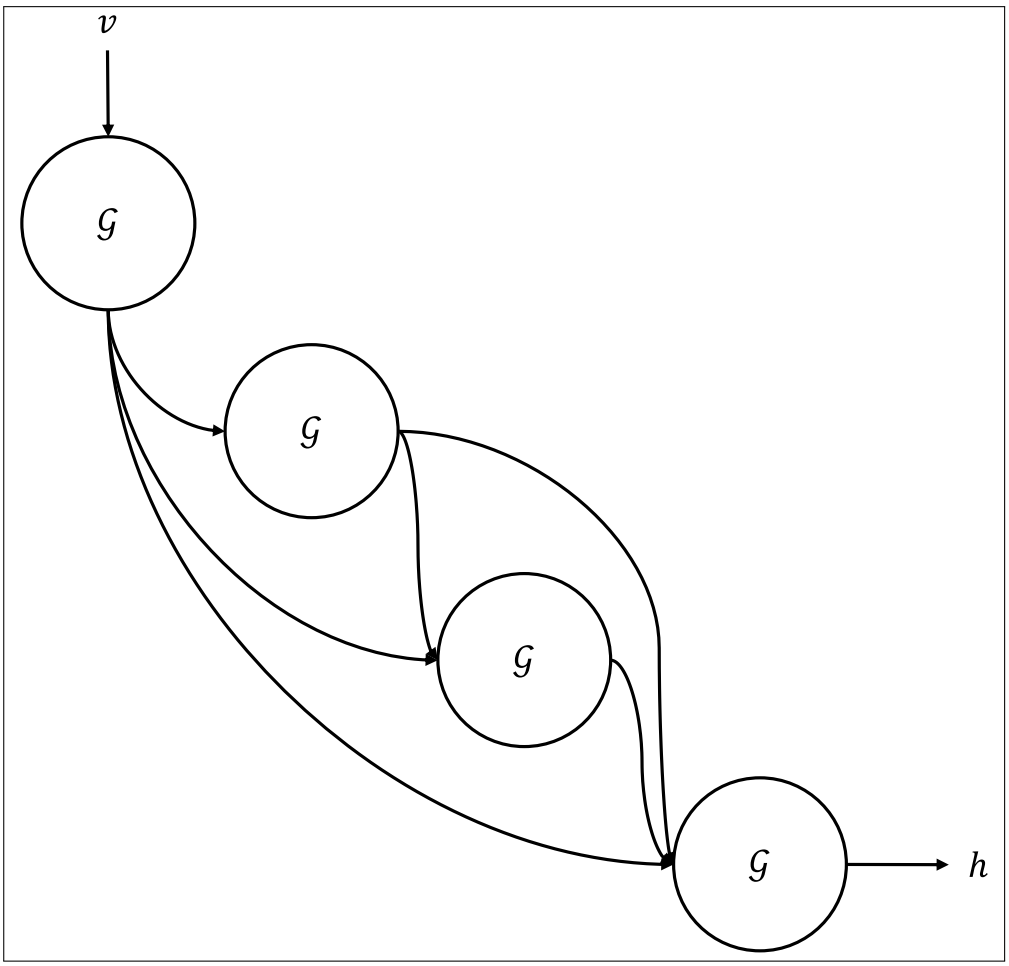

FIGURE E.2 – Schématisation de DenseNet [248].

améliorée présentée dans le document [266] :

$$y[t+1] = y[k] + \Delta_k(a\, f(y[k]) + b\, f(y[k] + \beta\Delta_k a\, f(y[k]))) \tag{E.18}$$

avec $y[k+1]$ la solution à $k+1$ de l'intégration de l'EDO, Δ_k le temps entre deux échantillons, a, b et β des paramètres à sélectionner selon la méthode d'intégration. La formulation de FractalNet pour un ordre 2 est[127] :

$$h[k] = \kappa_1 * v[k] + \kappa_2 * (\kappa_3 * v[k] + \mathcal{F}_1(v[k])) + \mathcal{F}_2(\kappa_3 * v[k] + f_1(v[k])) \tag{E.19}$$

avec $h[k]$ la solution en sortie de la couche FractalNet considérée, $\mathcal{F}$ la transformation neuronale considérée et $v[k]$ l'entrée de la couche FractalNet. $*$ est la convolution et κ_i sont des paramètres entrainables.

E.2.4 LM-ResNet

Nous pouvons aussi indiquer l'architecture de réseau ainsi que le *Linear multi-step neural network* LM-ResNet [127] relié à la méthode de résolution numérique multi-pas [125]. La formulation formulation mathématique de la méthode d'intégration multi-pas avec la méthode Adams pour un ordre deux est [125] :

$$y[k+1] = y[k] + \Delta_k \frac{3}{2} f(y[k]) - \Delta_k \frac{1}{2} f(y[k-1]) \tag{E.20}$$

avec $y[k+1]$ la solution à $k+1$ de l'intégration de l'EDO, Δ_k le temps entre deux échantillons. Par ailleurs, LM-ResNet nécessite une architecture profonde avec plusieurs couches cachées, la formulation de LM-ResNet [127] est :

$$h[k]^{l+1} = (1-\kappa^l)h[k]^l + \kappa^l h[k]^{l-1} + \mathcal{G}^l(h[k]^l) \tag{E.21}$$

avec $h[k]^{l+1}$ la sortie de LM-ResNet pour la cellule considérée, k^l un paramètre entrainable pour chaque couche cachée de LM-ResNet, $\mathcal{G}^l$ la transformation neuronale non linéaire considérée pour la couche l.

E.2.5 OdeNet

Ordinary differential equation neural network (OdeNet) est relié à une méthode de résolution numérique d'équation différentielle [267], à la place d'utiliser une séquence discrète, le réseau de neurones constitue une EDO. Durant l'apprentissage et l'inférence, le réseau est discrétisé à l'aide d'une méthode de résolution numérique. De ce faite, il est possible de sélectionner chacune des méthodes d'intégration implémentées dans l'outil informatique. La formulation mathématique de l'OdeNet est une EDO :

$$\dot{h}(t) = \mathcal{G}(h(t)) \tag{E.22}$$

avec $\dot{h}(t)$ la dérivée, puis $\mathcal{G}$ la transformation neuronale non linéaire. Ce réseau de neurones nécessite à chaque itération une l'application d'une méthode de résolution numérique de l'équation différentielle, ainsi l'apprentissage est allongé. Une librairie informatique est nécessaire, nous pouvons indiquer DiffEqFlux.jl [268] qui permet l'utilisation de ce réseau.

Annexe F

Modélisation détaillée de la serre agricole

Cette annexe présente en détail la modélisation de la serre agricole.

F.1 Bilan de puissance thermique

Le bilan de puissance thermique de la serre agricole est [204] :

$$\boxed{C_{air}\dot{T}_{air} = Q_{gh} + Q_{sol} + Q_{lamp} - Q_{convd} - Q_{rad} - Q_{trans} - Q_{ventil},\ (^{\circ}Cs^{-1})} \tag{F.1}$$

avec C_{air} la capacité thermique de l'air (JK^{-1}), Q_{gh} la puissance de chauffage de la serre (W), Q_{sol} la puissance du rayonnement solaire (W), Q_{lamp} la puissance thermique dégagée par les luminaires (W), Q_{convd} la puissance thermique de perte par la convection et la conduction de l'enveloppe en verre (W), Q_{rad} la puissance de perte radiative (W), Q_{trans} la puissance thermique absorbée par la plante lors de sa transpiration (W) et Q_{ventil} la perte thermique par la ventilation (W).

La capacité thermique de l'air est :

$$C_{air} = \rho_{air} V_{air} C_{p_{air}},\ (JK^{-1}) \tag{F.2}$$

avec ρ_{air} la densité de l'air kgm^{-3}, V_{air} le volume de la serre m^3 et $C_{p_{air}}$ la capacité thermique spécifique de l'air $Jkg^{-1}K^{-1}$.

La puissance de chauffage de la serre agricole est [217] :

$$Q_{gh} = \alpha_{gh}(T_{tuyau} - T_{air}),\ (W) \tag{F.3}$$

avec α_{gh} le coefficient de transfert thermique du réseau de chauffage $(W^{\circ -1})$, T_{tuyau} la tempé-

rature du fluide circulant dans le circuit de chauffage ($°C$) et T_{air} la température de l'air de la serre ($°C$).

La puissance du rayonnement solaire est :

$$Q_{sol} = \tau_a(A_{gh}I_b + A_sI_d),\ (W) \tag{F.4}$$

avec τ_a est la transmittance des matériaux de la serre agricole (–), A_{gh} est la surface de la serre (m^2), I_b est le rayonnement solaire direct (Wm^{-2}), A_s est la surface de l'enveloppe de la serre (m^2), I_d est le rayonnement solaire diffus (Wm^{-2}).

La puissance thermique dégagée par les lampes est :

$$Q_{lamp} = \eta_{th}P_e\frac{f_{on}}{100},\ (W) \tag{F.5}$$

avec η_{th} le ratio de puissance électrique transformée en chaleur (–), P_e est la puissance électrique des lampes (W), f_{on} est la fraction des lampes qui sont allumées.

La puissance thermique relâchée dans l'atmosphère par la conduction et la convection de la serre est :

$$Q_{convd} = A_sK_gK_c(T_{air} - T_{out}),\ (W) \tag{F.6}$$

avec A_s la surface de l'enveloppe de la serre, K_g est le coefficient de transfert thermique, K_c le coefficient d'ouverture ou de fermeture de l'écran thermique (–), T_{air} la température de l'air de la serre ($°C$) et T_{out} est la température extérieure ($°C$).

La puissance radiative émise et perdue par la serre est :

$$Q_{rad} = \epsilon_{12}A_s\sigma((T_{air} + 273.15)^4 - (T_{ciel} + 273.15)^4),\ (W) \tag{F.7}$$

avec ϵ_{12} le coefficient d'émission entre l'enveloppe et le ciel (–), A_s est la surface de l'enveloppe (m^2), σ est la constante de Stefan-Boltzmann ($Wm^{-2}°C^{-4}$), T_{air} est la température de l'air de la serre (T_{air}), T_{ciel} est la température du ciel ($°C$).

La température du ciel est égale à :

$$T_{ciel} = 94 + 12.6ln(e_0) - 13K_s + 0.341(T_{ext} + 273.15) - 273.15,\ (°C) \tag{F.8}$$

avec e_0 la valeur absolut de la pression vapeur de l'air (Pa) , K_s est le coefficient de clarté du ciel (–) et T_{ext} est la température extérieure à la serre agricole ($°C$).

Le coefficient d'émission est :

$$\epsilon_{12} = \frac{1}{\frac{1}{\epsilon_1} + \frac{1}{\epsilon_2} + 1}, \ (-) \tag{F.9}$$

avec ϵ_1 le coefficient d'émissivité de l'enveloppe de la serre (–) et ϵ_2 le coefficient d'émissivité du ciel (–).

La puissance thermique résultant de la transpiration des plantes est :

$$Q_{trans} = g_{plante} A_{gh} L(\mathtt{X}_{culture} - \mathtt{X}_{air}), \ (W) \tag{F.10}$$

avec g_{plante} est la conductance de transpiration de la plante (ms^{-1}), A_{gh} est la surface de la serre (m^2), L est l'énergie nécessaire pour l'évaporation de l'eau à la plante (Jg^{-1}), $\mathtt{X}_{culture}$ est la valeur absolue de la concentration en vapeur d'eau de l'air au niveau de la plante (gm^{-3}) et $\mathtt{X}_{air}$ est la valeur absolue de la concentration en vapeur d'eau de l'air dans la serre agricole gm^{-3}.

La conductance de la transpiration de la plante est :

$$g_{plante} = \frac{2LAI}{(1+\epsilon)r_b + r_s}, \ (ms^{-1}) \tag{F.11}$$

avec LAI l'indice de surface foliaire ($m^2 leaf\, m^{-2} sol$), ϵ est le ratio entre la chaleur sensible et latente de l'air saturé (–), r_b est la résistance de la couche limite : $200 sm^{-1}$ pour la production de tomate [269] et r_s est la résistance de stomatal (sm^{-1}). LAI dépend du développement de la plante et de la température, pour le cas d'une production de tomate, LAI peut être simplifié par la valeur constante de 3 [270].

Dans les plages de température d'une serre agricole, ϵ est approximé par [269] :

$$\epsilon = 0.7584 e^{0.0518 T_{air}}, \ (-) \tag{F.12}$$

avec T_{air} la température de la serre agricole ($^\circ C$).

La résistance de stomatal est :

$$r_s = (82 + 570 e^{-\gamma \frac{I_{rad_{plante}}}{LAI}})(1 + 0.023(T_{air} - 24.5)^2) \tag{F.13}$$

avec γ est un paramètre dépendant de la culture et de 0.4 pour la tomate [218], $I_{rad_{plante}}$ le rayonnement solaire au niveau de la plante (Wm^{-2}), LAI l'indice de surface foliaire ($m^2 leaf\, m^{-2} sol$) et T_{air} la température de l'air de la serre agricole ($^\circ C$).

La rayonnement solaire au niveau de la plante est :

$$I_{rad_{plante}} = 0.86(1 - e^{-0.7LAI})(q_{sol} + P_e \frac{f_{on}}{100}),\ (Wm^{-2}) \tag{F.14}$$

avec LAI l'indice de surface foliaire ($m^2 leaf\, m^{-2} sol$), Q_{sol} est la puissance du rayonnement solaire $q_{qol} = Q_{sol}/A_{gh}$, P_e est la puissance électrique des lampes (Wm^{-2}) et f_{on} la fraction de lampes allumées (–).

La valeur absolue de la concentration en vapeur d'eau de l'air au niveau de la plante est :

$$\mathrm{X}_{plante} = \mathrm{X}_{air_{sat}} + \epsilon \frac{r_b}{2LAI} + \frac{I_{rad_{plante}}}{L},\ (gm^{-3}) \tag{F.15}$$

avec $\mathrm{X}_{air_{sat}}$ la concentration de vapeur saturée (gm^{-3}), ϵ le ratio entre la chaleur sensible et latente de l'air saturé, Eq.F.12, r_b la résistance de la couche limite, $200 sm^{-1}$ pour la production de tomate [269], LAI l'indice de surface foliaire $m^2 leaf\, m^{-2} sol$, $I_{rad_{plante}}$ le rayonnement solaire au niveau de la plante et L l'énergie nécessaire pour l'évaporation de l'eau à la plante (Jg^{-1}).

La concentration de vapeur saturée est approximée entre $10°C$ et $40°C$ par :

$$\mathrm{X}_{air_{sat}} = 5.5638 e^{0.0572 T_{air}},\ (gm^{-3}) \tag{F.16}$$

avec T_{air} la température de la serre ($°C$).

L'énergie perdue par la ventilation résulte de la vitesse du vent, de la différence de température entre la serre et l'extérieur ainsi que de l'ouverture des fenêtres, telle que :

$$Q_{ventil} = g_v \rho_{air} C_{p_{air}} A_{gh} (T_{air} - T_{ext}),\ (W) \tag{F.17}$$

avec g_v la vitesse de la ventilation (ms^{-1}), ρ_{air} la densité de l'air (kgm^{-3}), la capacité thermique spécifique de l'air ($Jkg^{-1}K^{-1}$), A_{gh} la surface de la serre (m^2), T_{air} la température de l'air de la serre ($°C$) et T_{ext} la température de l'air extérieur ($°C$).

La vitesse de l'air résultant de la ventilation est :

$$g_v = \frac{1}{2} f r_{fenêtre} \sqrt{\phi_{vent}^2 + \phi_{temp}^2},\ (ms^{-1}) \tag{F.18}$$

avec $fr_{fenêtre}$ le nombre de fenêtres par unité de surface : 0.078 $fenêtre.m^{-2}$ [271], ϕ_{vent} le volume d'air déplacé par le vent extérieur $m^3 s^{-1}$, ϕ_{temp} le volume d'air déplacé par la différence de la température entre l'air de la serre et l'air extérieur ($m^3 s^{-1}$).

Le volume d'air déplacé par le vent extérieur est :

$$\phi_{vent} = uA_0G_lG_w,\ (m^3s^{-1}fenêtre^{-1}) \tag{F.19}$$

avec u la vitesse du vent extérieur (ms^{-1}), A_0 est la surface d'une fenêtre (m^2), G_l coefficient de la fenêtre sous le vent (–) et G_w coefficient de la fenêtre au vent.

Les coefficients de la fenêtre au vent et sous le vent sont :

$$G_l = 2.29 \times 10^{-2}(1 - e^{-\frac{\theta}{21.1}}),\ (-) \tag{F.20}$$

$$G_w = 1.2 \times 10^{-3}\theta e^{\frac{\theta}{211}},\ (-) \tag{F.21}$$

avec θ l'angle d'ouverture des fenêtres de la serre (°).

Le volume d'air déplacé par la différence de la température entre l'air de la serre et l'air extérieur est :

$$\phi_{temp} = C_f\frac{l}{3}\sqrt{|g\beta\Delta T|}H^{1.5},\ (m^3s^{-1}fenêtre^{-1}) \tag{F.22}$$

avec C_f paramètre de la décharge d'énergie résultant de l'ouverture de la fenêtre : 0.6 (–) [271], l est la longueur des fenêtres (m), g est l'accélération gravitationnelle ms^{-2}, β est le coefficient d'expansion thermique : $\frac{1}{283}K^{-1}$ [271] et H est la hauteur d'ouverture de la fenêtre (m), Fig. F.1.

La hauteur d'ouverture de la fenêtre H est :

$$H = h(sin(\psi) - sin(\psi - \theta)), (m) \tag{F.23}$$

avec ψ l'angle du toit ° et θ l'angle d'ouverture de la fenêtre (°).

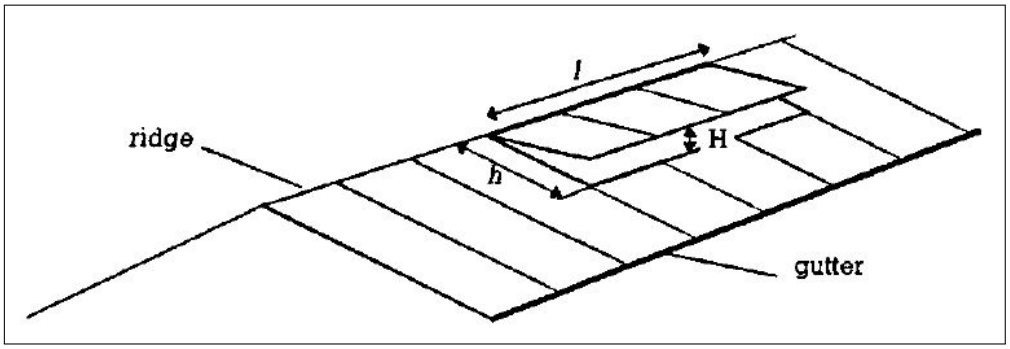

FIGURE F.1 – Illustration de H [271].

F.2 Bilan d'humidité

Le bilan d'humidité de la serre agricole est [218] :

$$\boxed{h\dot{\mathtt{X}}_{air} = \phi_{\mathtt{X}_{trans}} - \phi_{\mathtt{X}_{couv}} - \phi_{\mathtt{X}_{desh}} - \phi_{\mathtt{X}_{vent}},\ (gm^{-3}s^{-1})} \tag{F.24}$$

avec h est la hauteur de la serre agricole (m), $\phi_{\mathtt{X}_{trans}}$ est vapeur d'eau qui est générée par les plantes avec leurs transpirations ($gm^{-2}s^{-1}$), $\phi_{\mathtt{X}_{couv}}$ est la condensation sur les vitres de la serre agricole ($gm^{-2}s^{-1}$), $\phi_{\mathtt{X}_{desh}}$ est la condensation réalisée par le déshumidificateur ($gm^{-2}s^{-1}$) et $\phi_{\mathtt{X}_{vent}}$ est le flux de vapeur résultant de l'action de la ventilation ($gm^{-2}s^{-1}$).

La vapeur occasionnée par la transpiration de la plante est [218], :

$$\phi_{\mathtt{X}_{trans}} = \frac{Q_{trans}}{A_{gh}L},\ (gm^{-2}s^{-1}) \tag{F.25}$$

avec Q_{trans} de l'Eq.F.10, A_{gh} la surface de la serre m^2 et L l'énergie nécessaire pour l'évaporation de l'eau (Jg^{-1}).

La condensation sur les vitres de la serre est :

$$\phi_{\mathtt{X}_{couv}} = g_c(0.2522e^{0.0485T_{air}}(T_{air} - T_{ext}) - (\mathtt{X}_{sat} - \mathtt{X}_{air})),\ (gm^{-2}s^{-1}) \tag{F.26}$$

avec g_c la conductance de la condensation (ms^{-1}), T_{air} la température de l'air de la serre agricole ($^\circ C$), T_{ext} la température extérieure à la serre agricole ($^\circ C$), $\mathtt{X}_{sat}$ la concentration de vapeur de l'air saturé (gm^{-3}) et $\mathtt{X}_{air}$ la concentration en eau de l'air de la serre agricole (gm^{-3}).

La conductance de la condensation est :

$$g_c = max(0, p_{gc}\sqrt[3]{T_{air} - T_{couv}}) \tag{F.27}$$

avec p_{gc} un paramètre de propriété de condensation de la surface, $1.8 \times 10^3 m^\circ C^{-\frac{1}{3}} s^{-1}$, T_{air} est la température de l'air de la serre agricole $^\circ C$ et T_{couv} est la température de l'enveloppe de la serre agricole.

La condensation résultant du déshumidificateur de la serre agricole est :

$$\phi_{\mathtt{X}_{desh}} = g_{c_{desh}}(\mathtt{X}_{planque} - \mathtt{X}_{air}),\ (gm^{-2}s^{-1}) \tag{F.28}$$

avec $g_{c_{desh}}$ la conductance de la condensation (ms^{-1}), $\mathtt{X}_{planque}$ est la concentration en eau saturée

à la plaque de l'échangeur de l'Eq.F.16 avec le remplacement de la température de l'air par la température de la planque et $\mathtt{X}_{air}$ la concentration en eau de l'air de la serre (gm^{-3}).

La conductance de la condensation est :

$$g_{c_{desh}} = max(0, p_{g_{c_{desh}}} \sqrt[3]{T_{air} - T_{plaque}}),\ (ms^{-1}) \tag{F.29}$$

avec $p_{g_{c_{desh}}}$ un paramètre de propriété de condensation de la surface, $0.25 \times 10^3 m°C^{-\frac{1}{3}} s^{-1}$, T_{air} la température de l'air de la serre agricole ($°C$), T_{plaque} la température de la plaque de l'échangeur ($°C$).

Le flux de vapeur occasionné par la ventilation est :

$$\phi_{\mathtt{X}_{vent}} = g_v(\mathtt{X}_{air} - \mathtt{X}_{ext}),\ (gm^{-2}s^{-1}) \tag{F.30}$$

avec g_v le débit de la ventilation (ms^{-1}), $\mathtt{X}_{air}$ la concentration en eau de l'air de la serre (gm^{-3}) et $\mathtt{X}_{ext}$ la concentration en eau de l'air extérieur (gm^{-3}).

F.3 Bilan du dioxyde de carbone

Le bilan du CO_2 de la serre agricole est [217] :

$$\boxed{h\dot{CO}_{2_{air}} = \phi_{CO_{2_{inj}}} - \phi_{CO_{2_{ass}}} - \phi_{CO_{2_{vent}}},\ (gm^{-3}s^{-1})} \tag{F.31}$$

avec h la hauteur de la serre agricole (m), $\phi_{CO_{2_{inj}}}$ est l'injection de CO_2 dans la serre agricole ($gm^{-2}s^{-1}$), $\phi_{CO_{2_{ass}}}$ est le CO_2 assimilé par les plantes ($gm^{-2}s^{-1}$) et $\phi_{CO_{2_{vent}}}$ est l'échange de CO_2 entre la serre et l'air extérieur occasionné par la ventilation.

L'assimilation du CO_2 par les plantes est :

$$\phi_{CO_{2_{ass}}} = 2.2 \times 10^{-3} \frac{1}{1 + \frac{0.42}{\mathtt{CO}_{2_{air}}}} (1 - e^{-0.003 I_{rad_{plante}}}),\ (gm^{-2}s^{-1}) \tag{F.32}$$

avec $\mathtt{CO}_{2_{air}}$ la concentration en CO_2 dans la serre agricole, $I_{rad_{plante}}$ le rayonnement solaire au niveau de la plante (Wm^{-2}).

L'échange de CO_2 avec l'extérieur est :

$$\phi_{CO_{2_{vent}}} = g_v(\mathtt{CO}_{2_{air}} - CO_{2_{ext}}),\ (gm^{-2}s^{-1}) \tag{F.33}$$

avec g_v est le débit de ventilation (ms^{-1}), $\mathtt{CO}_{2_{\mathtt{air}}}$ la concentration en CO_2 dans la serre agricole (gm^{-3}) et $CO_{2_{ext}}$ la concentration en CO_2 extérieur en (gm^{-3}).

F.4 Stockage thermique

Les serres agricoles possèdent des stockages de chaleur. Pour une modélisation simplifiée avec un mélange homogène de la température de l'eau, un noeud du stockage thermique, le modèle de bilan de puissance est [229] :

$$\boxed{C_{tes}\dot{T}_{tes} = Q_{heat} - Q_{load} - Q_{loss},\ (^{\circ}Cs^{-1})} \tag{F.34}$$

avec C_{tes} la capacité thermique du fluide du stockage de la chaleur (JK^{-1}), T_{tes} la température de stockage ($^{\circ}C$), Q_{heat} la puissance de chauffage (W), Q_{load} la puissance soutirée (W) et Q_{loss} la puissance perdue (W).

La puissance de chauffage est :

$$Q_{heat} = \dot{m}_{heat}C_{p_{tes}}(T_{heat} - T_{tes}),\ (W) \tag{F.35}$$

avec $\dot{m}_{heat}$ le débit massique de chargement (kgs^{-1}), $C_{p_{tes}}$ la capacité thermique massique du fluide de stockage ($JK^{-1}kg^{-1}$), T_{heat} la température de la source chaude ($^{\circ}C$) et T_{tes} la température du stockage ($^{\circ}C$).

La puissance soutirée au stockage de chaleur est :

$$Q_{load} = \dot{m}_{load}C_{p_{tes}}(T_{tes} - T_{main}),\ (W) \tag{F.36}$$

avec $\dot{m}_{load}$ le débit massique de soutirage (kgs^{-1}), $C_{p_{tes}}$ la capacité thermique massique du fluide de stockage ($JK^{-1}kg^{-1}$), T_{main} la température de la source froide ($^{\circ}C$) et T_{tes} la température du stockage ($^{\circ}C$).

La puissance perdue par échange avec l'extérieur est :

$$Q_{loss} = UA(T_{ext} - T_{tes}),\ (W) \tag{F.37}$$

avec UA le coefficient d'échange de chaleur ($K^{-1}W$), T_{ext} est la température extérieure ($^{\circ}C$) et T_{tes} est la température du stockage ($^{\circ}C$).

Il arrive qu'un stockage de chaleur présente une stratification thermique (la température

en haut du stockage est plus importante qu'à la base du stockage). Le modèle nodal n'est plus valable et il est nécessaire d'utiliser un modèle multinœuds. La modélisation est améliorée par une discrétisation en plusieurs couches i du stockage thermique [229], [272] :

$$M_i C_{p_{tes}} \dot{T}_i = \begin{cases} (\dot{m}_{heat} - \dot{m}_{load}) C_{p_{tes}} (T_{i-1} - T_i) - UA(T_{ext} - T_i), & si\ \dot{m}_{heat} \geq \dot{m}_{load} \\ (\dot{m}_{load} - \dot{m}_{heat}) C_{p_{tes}} (T_{i-1} - T_i) - UA(T_{ext} - T_i), & si\ \dot{m}_{heat} < \dot{m}_{load} \end{cases} \tag{F.38}$$

avec M_i la masse du fluide de stockage dans la couche i (kg), $C_{p_{tes}}$ la capacité thermique spécifique du fluide de stockage ($JK^{-1}kg^{-1}$), T_i la température dans la couche i ($°C$), $\dot{m}_{heat}$ le débit massique de chargement (kgs^{-1}), $\dot{m}_{load}$ le débit massique de soutirage (kgs^{-1}), UA le coefficient d'échange de chaleur ($K^{-1}W$) et T_{ext} la température extérieure ($°C$).

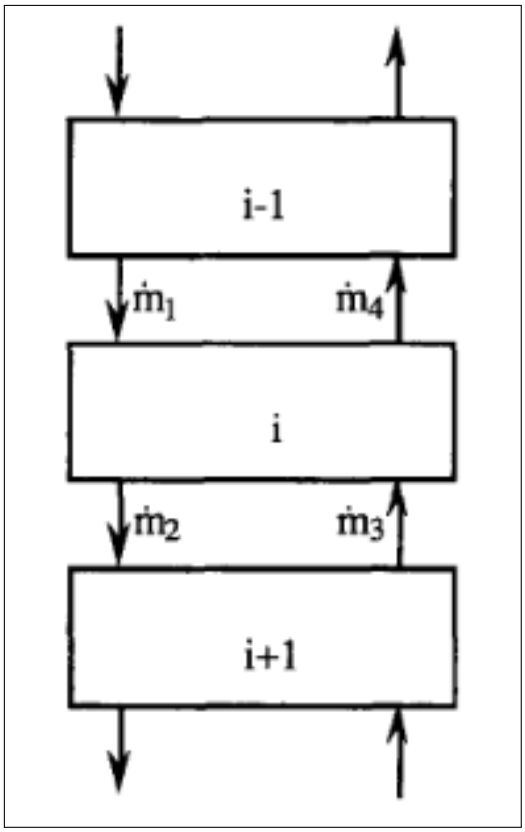

FIGURE F.2 – Modèle multinœud d'un stockage thermique [273].

F.5 Unité de production de chaleur

Tout d'abord, il est considéré que la serre agricole possède une unité de production chaleur. Pour simplification la chaudière est séparée en trois étages, gaz, solide et liquide, une illustration est visible en Fig.F.3.

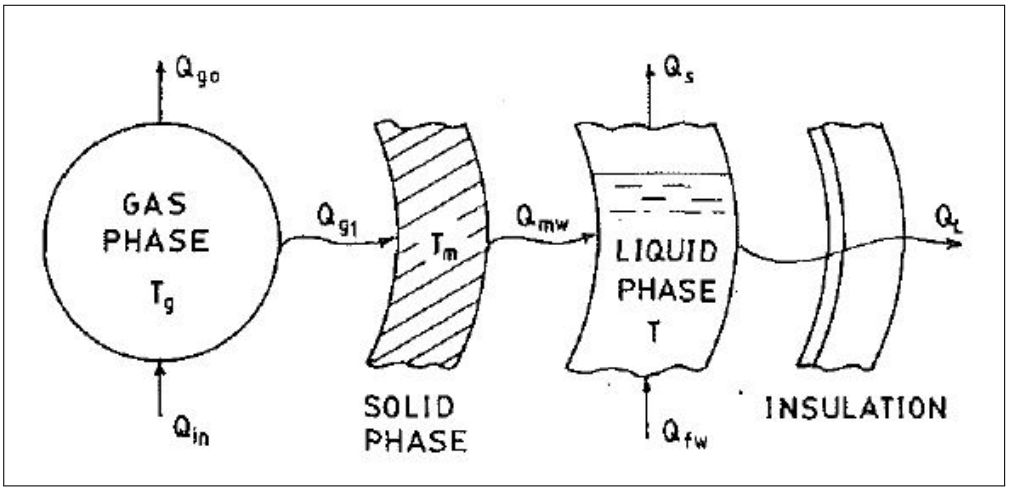

FIGURE F.3 – Trois étages pour la modélisation de la chaudière [230].

Le bilan énergétique d'une chaudière simplifiée est [230] est :

$$\boxed{C_{boil}\dot{T}_m = Q_{in} - Q_{go} - Q_e - Q_{loss},\ (^\circ C s^{-1})} \tag{F.39}$$

avec C_{boil} la capacité thermique de la chaudière (masse de l'échangeur) (Jk^{-1}), Q_{in} la puissance résultant de la combustion (W), Q_{go} la puissance perdue dans les gaz d'échappement (W), Q_e la puissance exportée par le caloporteur de la chaudière (W) et Q_{loss} la puissance perdue dans l'atmosphère (W).

La puissance de combustion est évaluée par :

$$Q_{in} = \dot{m}_f[H_f + C_f(T_f - T_o)],\ (W) \tag{F.40}$$

avec $\dot{m}_f$ le débit massique du carburant (kgs^{-1}), H_f le pouvoir calorifique du carburant $(KJkg^{-1})$, C_f la capacité thermique du carburant $(kJkg^{-1\circ}C^{-1})$, T_f température du carburant $(^\circ C)$ et T_o température *datum state* $(^\circ C)$.

La puissance perdue dans les gaz de combustion est :

$$Q_{go} = m_f[\phi_a A_0 C_{go}(T_{go} - T_o) + G_{wf}\Delta H + G_{wa}C_s(T_{go} - T_o) + 30.5\phi_a A_0(CO)],\ (W) \tag{F.41}$$

avec $\dot{m}_f$ le débit massique du carburant (kgs^{-1}), ϕ_a le ratio du débit d'air de combustion entre l'actuel et le théorique (–), A_0 le débit d'air théorique par unité de carburant (Nm^3kg^{-1}), C_{go} la capacité thermique des gaz d'échappement ($kJkg^{-1}{}^\circ C^{-1}$), T_{go} la température des gaz d'échappement $^\circ C$, T_o la température de l'état datum $^\circ C$, G_{wf} la formation de l'eau dans la combustion, ΔH la chaleur latente de la vapeur d'eau $kcalkg^{-1}oil$, G_{wa} la concentration de vapeur d'eau de l'air ($kgvapeurkg^{-1}oil$), C_s la capacité thermique de la vapeur ($kJkg^{-1}{}^\circ C$).

La formation de l'eau dans la combustion est :

$$G_{wf} = 0.09h + 0.01w \tag{F.42}$$

avec h et w la composition d'hydrogène et de l'eau dans le combustible.

La chaleur latente de la vapeur d'eau est :

$$\Delta H = 597.5 + C_s(T_{go} - T_o),\ (kcalkg^{-1}oil) \tag{F.43}$$

avec C_s la capacité thermique de la vapeur ($kJkg^{-1}{}^\circ C$), T_{go} la température des gaz d'échappement $^\circ C$ et T_o la température de l'état datum $^\circ C$.

La concentration de vapeur d'eau de l'air est :

$$G_{wa} = 1.293\gamma_a\phi_a A_0,\ (kg\ vapeur\ kg^{-1}oil) \tag{F.44}$$

avec γ_a l'humidité absolue de l'air de combustion ($kgvapeurkg^{-1}oil$), ϕ_a le ratio du débit d'air de combustion entre l'actuel et le théorique (–), A_0 le débit d'air théorique par unité de carburant (Nm^3kg^{-1}).

Le ratio du débit d'air de combustion entre l'actuel et le théorique est :

$$\phi_a = \frac{21}{21 - 79(\frac{O_2 - 0.5CO}{N_2})},\ (-) \tag{F.45}$$

Le débit d'air théorique par unité de carburant est :

$$A_0 = 0.0889c + 0.267(h - \frac{o}{8}) + 0.333s,\ (Nm^3kg^{-1}) \tag{F.46}$$

avec c la concentration en carbon, o la concentration en oxygène et s la concentration en sulfure.

La puissance exportée par le caloporteur de la chaudière est :

$$Q_e = \dot{m}_w C_{p_w}(T_{wo} - T_{fw}),\ (W) \tag{F.47}$$

avec $\dot{m}_w$ le débit massique de l'eau d'alimentation kgs^{-1}, C_{p_w} la capacité thermique massique du fluide $JK^{-1}kg^{-1}$, T_{wo} la température de sortie ($^\circ C$) et T_{fw} la température d'alimentation en eau ($^\circ C$).

La puissance thermique perdue dans l'atmosphère est :

$$Q_{loss} = h_{wa}A_{wa}(T_{wo} - T_{air}),\ (W) \tag{F.48}$$

avec h_{wa} le coefficient de transfert thermique ($Wm^{-2\circ}C^{-1}$), A_{wa} la surface d'échange avec l'air ambiant (m^2), T_{wo} la température de sortie ($^\circ C$) et T_{air} l'air ambiant externe à la chaudière ($^\circ C$).

Sixième partie

Références bibliographiques

BIBLIOGRAPHIE

[1] C. S. Mattick, E. Williams et B. R. Allenby, « Historical trends in global energy consumption », *IEEE Technology and Society Magazine*, **29**, *3*, p. 22-30, 2010. doi : 10.1109/MTS.2010.938106.

[2] T. Ahmad et D. Zhang, « A critical review of comparative global historical energy consumption and future demand : The story told so far », *Energy Reports*, **6**, p. 1973-1991, 2020. doi : 10.1016/j.egyr.2020.07.020.

[3] J. F. Coste, A. Coursimault, F. Brissaud, J. Bongrand et D. Chauvin, « Les prévisions du GIEC », in *Changement climatique*. EDP Sciences, 2018, p. 21-24. doi : 10.1051/978-2-7598-2250-8-005.

[4] C. Fauvel, F. Claveau, P. Chevrel et P. Fiani, « A flexible design methodology to solve energy management problems », *International Journal of Electrical Power & Energy Systems*, **97**, p. 220-232, 2018, issn : 0142-0615. doi : 10.1016/j.ijepes.2017.11.005.

[5] L. M. Serra, M.-A. Lozano, J. Ramos, A. V. Ensinas et S. A. Nebra, « Polygeneration and efficient use of natural resources », *Energy*, **34**, *5*, p. 575-586, 2009, 4th Dubrovnik Conference. doi : 10.1016/j.energy.2008.08.013.

[6] P. Mancarella, « MES (multi-energy systems) : An overview of concepts and evaluation models », *Energy*, **65**, p. 1-17, 2014. doi : 10.1016/j.energy.2013.10.041.

[7] F. Lamnabhi-Lagarrigue, A. Annaswamy, S. Engell et al., « Systems & Control for the future of humanity, research agenda : Current and future roles, impact and grand challenges », *Annual Reviews in Control*, **43**, p. 1-64, 2017. doi : 10.1016/j.arcontrol.2017.04.001.

[8] T. Samad et A. Annaswamy, *The impact of control technology : Overview, success stories, and research challenges*. IEEE Control Systems Society, 2011. adresse : http://ieeecss.org/impact-control-technology-2nd-edition.

[9] M. Yue, H. Lambert, E. Pahon, R. Roche, S. Jemei et D. Hissel, « Hydrogen energy systems : A critical review of technologies, applications, trends and challenges », *Renewable and Sustainable Energy Reviews*, **146**, p. 111 180, 2021. doi : 10.1016/j.rser.2021.111180.

[10] S. Kuboth, F. Heberle, A. König-Haagen et D. Brüggemann, « Economic model predictive control of combined thermal and electric residential building energy systems », *Applied Energy*, **240**, p. 372-385, 2019. doi : 10.1016/j.apenergy.2019.01.097.

[11] G. T. Ayele, M. T. Mabrouk, P. Haurant, B. Laumert et B. Lacarrière, « Optimal placement and sizing of heat pumps and heat only boilers in a coupled electricity and heating networks », *Energy*, **182**, p. 122-134, 2019. doi : 10.1016/j.energy.2019.06.018.

[12] X. Xu, H. Jia, D. Wang, D. C. Yu et H.-D. Chiang, « Hierarchical energy management system for multi-source multi-product microgrids », *Renewable Energy*, **78**, p. 621-630, 2015. doi : 10.1016/j.renene.2015.01.039.

[13] P. Báez-González, A. J. del Real, M. A. Ridao Carlini et C. Bordons, « Day-ahead economic optimization of energy use in an olive mill », *Control Engineering Practice*, **54**, p. 91-103, 2016. doi : 10.1016/j.conengprac.2016.05.019.

[14] P. Albertos et I. Mareels, « Introduction », in *Feedback and control for everyone*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2010, p. 3-27. doi : 10.1007/978-3-642-03446-6_1.

[15] K. Åström et T. Hägglund, « The future of PID control », *Control Engineering Practice*, **9**, *11*, p. 1163-1175, 2001. doi : 10.1016/S0967-0661(01)00062-4.

[16] Y. Granjon, *Automatique : systèmes linéaires, non linéaires, à temps continu, à temps*. Dunod, 2001.

[17] S. Clarke et C. Ayers, « Current developments in computer hierarchies for industrial control », *Radio and Electronic Engineer*, **38**, *1*, p. 5-12, 1969. doi : 10.1049/ree.1969.0068.

[18] W. Aspray, « The Intel 4004 microprocessor : what constituted invention ? », *IEEE Annals of the History of Computing*, **19**, *3*, p. 4-15, 1997. doi : 10.1109/85.601727.

[19] M. M. Waldrop, « More than moore », *Nature*, **530**, *7589*, p. 144-148, 2016. doi : 10.1038/530144a.

[20] F. Claveau, « Contribution à l'analyse et la commande structurée des grands systèmes », thèse de doct., Ecole Centrale de Nantes, 2005. adresse : hal:tel-00087765.

[21] R. Scattolini, « Architectures for distributed and hierarchical Model Predictive Control – A review », *Journal of Process Control*, **19**, *5*, p. 723-731, 2009. doi : 10.1016/j.jprocont.2009.02.003.

[22] A. Ulbig, M. Arnold, S. Chatzivasileiadis et G. Andersson, « Framework for multiple time-scale cascaded MPC application in power systems », *IFAC Proceedings Volumes*, **44**, *1*, p. 10 472-10 480, 2011, 18th IFAC World Congress. doi : 10.3182/20110828-6-IT-1002.01859.

[23] C. Vermillion, A. Menezes et I. Kolmanovsky, « Stable hierarchical model predictive control using an inner loop reference model and λ-contractive terminal constraint sets », *Automatica*, **50**, *1*, p. 92-99, 2014. doi : 10.1016/j.automatica.2013.10.009.

[24] B. Picasso, D. De Vito, R. Scattolini et P. Colaneri, « An MPC approach to the design of two-layer hierarchical control systems », *Automatica*, **46**, *5*, p. 823-831, 2010. doi : 10.1016/j.automatica.2010.02.013.

[25] A. C. Yu et D. Margoliash, « Temporal hierarchical control of singing in birds », *Science*, **273**, *5283*, p. 1871-1875, 1996. doi : 10.1126/science.273.5283.1871.

[26] P. Ruoff, M. Zakhartsev et H. V. Westerhoff, « Temperature compensation through systems biology », *The FEBS Journal*, **274**, *4*, p. 940-950, 2007. doi : 10.1111/j.1742-4658.2007.05641.x.

[27] C. Martin-Rios, « Innovation in organisational control systems : toward greater accountability », *International Journal of Business Performance Management*, **16**, *4*, p. 373-388, 2015. doi : 10.1504/IJBPM.2015.072235.

[28] J. Z. Florez, « Etudes de commande par décomposition-coordination pour l'optimisation de la conduite de vallées hydroélectriques », thèse de doct., Université de Grenoble, 2012. adresse : hal:tel-00745001.

[29] J. Zárate Flórez, J. Martinez, G. Besançon et D. Faille, « Decentralized-coordinated model predictive control for a hydro-power valley », *Mathematics and Computers in Simulation*, **91**, p. 108-118, 2013. doi : 10.1016/j.matcom.2012.04.005.

[30] J. Lai, Y. Song et X. Du, « Hierarchical coordinated control of flywheel energy storage matrix systems for wind farms », *IEEE/ASME Transactions on Mechatronics*, **23**, *1*, p. 48-56, 2018. doi : 10.1109/TMECH.2017.2654067.

[31] E. Camacho et A. Gallego, « Optimal operation in solar trough plants : A case study », *Solar Energy*, **95**, p. 106-117, 2013. doi : 10.1016/j.solener.2013.05.029.

[32] K. Edlund, J. Bendtsen et J. Jørgensen, « Hierarchical model-based predictive control of a power plant portfolio », *Control Engineering Practice*, **19**, *10*, p. 1126-1136, 2011. doi : 10.1016/j.conengprac.2011.06.002.

[33] S. Sen et V. Kumar, « Microgrid control : A comprehensive survey », *Annual Reviews in Control*, **45**, p. 118-151, 2018. doi : 10.1016/j.arcontrol.2018.04.012.

[34] C. Tsay et M. Baldea, « 110th anniversary : using data to bridge the time and length scales of process systems », *Industrial & Engineering Chemistry Research*, **58**, *36*, p. 16 696-16 708, 2019. doi : 10.1021/acs.iecr.9b02282.

[35] K. J. Åström et P. Kumar, « Control : A perspective », *Automatica*, **50**, *1*, p. 3-43, 2014. doi : 10.1016/j.automatica.2013.10.012.

[36] H. P. Geering, « Optimal control », in *Optimal control with engineering applications*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2007, p. 23-74. doi : 10.1007/978-3-540-69438-0_2.

[37] L. Saleh, P. Chevrel et J.-F. Lafay, « Optimal control with preview for lateral steering of a passenger car : Design and test on a driving simulator », in *Time delay systems : Methods, applications and new trends*, R. Sipahi, T. Vyhlídal, S.-I. Niculescu et P. Pepe, éd. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012, p. 173-185. doi : 10.1007/978-3-642-25221-1_13.

[38] « Receding horizon optimal control with constraints », in *Constrained control and estimation : An optimisation approach*. London : Springer London, 2005, p. 85-101. doi : 10.1007/1-84628-063-X_4.

[39] J. Richalet, A. Rault, J. Testud et J. Papon, « Model predictive heuristic control : Applications to industrial processes », *Automatica*, **14**, *5*, p. 413-428, 1978. doi : 10.1016/0005-1098(78)90001-8.

[40] A. Weiss, M. Baldwin, R. S. Erwin et I. Kolmanovsky, « Model predictive control for spacecraft rendezvous and docking : strategies for handling constraints and case studies », *IEEE Transactions on Control Systems Technology*, **23**, *4*, p. 1638-1647, 2015. doi : 10.1109/TCST.2014.2379639.

[41] B.-L. Ye, W. Wu, K. Ruan et al., « A survey of model predictive control methods for traffic signal control », *IEEE/CAA Journal of Automatica Sinica*, **6**, *3*, p. 623-640, 2019. doi : 10.1109/JAS.2019.1911471.

[42] J. Kabzan, L. Hewing, A. Liniger et M. N. Zeilinger, « Learning-based model predictive control for autonomous racing », *IEEE Robotics and Automation Letters*, **4**, *4*, p. 3363-3370, 2019. doi : 10.1109/LRA.2019.2926677.

[43] M. Ellis et P. D. Christofides, « Integrating dynamic economic optimization and model predictive control for optimal operation of nonlinear process systems », *Control Engineering Practice*, **22**, p. 242-251, 2014. doi : 10.1016/j.conengprac.2013.02.016.

[44] Amin, R. T. Bambang, A. S. Rohman, C. J. Dronkers, R. Ortega et A. Sasongko, « Energy management of fuel cell/battery/supercapacitor hybrid power sources using model predictive control », *IEEE Transactions on Industrial Informatics*, **10**, *4*, p. 1992-2002, 2014. doi : 10.1109/TII.2014.2333873.

[45] A. Lefort, R. Bourdais, G. Ansanay-Alex et H. Guéguen, « Hierarchical control method applied to energy management of a residential house », *Energy and Buildings*, **64**, p. 53-61, 2013. doi : 10.1016/j.enbuild.2013.04.010.

[46] J. B. Rawlings, N. R. Patel, M. J. Risbeck, C. T. Maravelias, M. J. Wenzel et R. D. Turney, « Economic MPC and real-time decision making with application to large-scale HVAC energy systems », *Computers & Chemical Engineering*, **114**, p. 89-98, 2018. doi : 10.1016/j.compchemeng.2017.10.038.

[47] F. Liberati et A. Di Giorgio, « Economic model predictive and feedback control of a smart grid prosumer node », *Energies*, **11**, *1*, 2018. doi : 10.3390/en11010048.

[48] B. Baeten, F. Rogiers et L. Helsen, « Reduction of heat pump induced peak electricity use and required generation capacity through thermal energy storage and demand response », *Applied Energy*, **195**, p. 184-195, 2017. doi : 10.1016/j.apenergy.2017.03.055.

[49] P. R. Mendes, L. V. Isorna, C. Bordons et J. E. Normey-Rico, « Energy management of an experimental microgrid coupled to a V2G system », *Journal of Power Sources*, **327**, p. 702-713, 2016. doi : 10.1016/j.jpowsour.2016.07.076.

[50] L. M. Ramírez-Elizondo et G. (Bob) Paap, « Scheduling and control framework for distribution-level systems containing multiple energy carrier systems : Theoretical approach and illustrative example », *International Journal of Electrical Power & Energy Systems*, **66**, p. 194-215, 2015. doi : 10.1016/j.ijepes.2014.10.045.

[51] X. Xu, H. Jia, D. Wang, D. C. Yu et H.-D. Chiang, « Hierarchical energy management system for multi-source multi-product microgrids », *Renewable Energy*, **78**, p. 621-630, 2015. doi : 10.1016/j.renene.2015.01.039.

[52] A. Jambagi, M. Kramer et V. Cheng, « Energy storage within a multi-energy carrier optimisation framework », in *2016 IEEE International Energy Conference (ENERGYCON)*, 2016, p. 1-6. doi : 10.1109/ENERGYCON.2016.7513885.

[53] S. Kuboth, F. Heberle, A. König-Haagen et D. Brüggemann, « Economic model predictive control of combined thermal and electric residential building energy systems », *Applied Energy*, **240**, p. 372-385, 2019. doi : 10.1016/j.apenergy.2019.01.097.

[54] R. P. Menon, F. Maréchal et M. Paolone, « Intra-day electro-thermal model predictive control for polygeneration systems in microgrids », *Energy*, **104**, p. 308-319, 2016. doi : 10.1016/j.energy.2016.03.081.

[55] M. Arnold, R. R. Negenborn, G. Andersson et B. De Schutter, « Model-based predictive control applied to multi-carrier energy systems », in *2009 IEEE Power Energy Society General Meeting*, 2009, p. 1-8. doi : 10.1109/PES.2009.5275230.

[56] F. Verhulst, « Mathematics is the art of giving the same name to different things : An interview with Henri Poincaré », *Nieuw archief voor wiskunde*, **5/13**, *3*, p. 154-158, 2012.

[57] P. Fritzson, « Introduction to modeling and simulation », in *Principles of object oriented modeling and simulation with Modelica 3.3*. John Wiley & Sons, Ltd, 2014, chap. 1, p. 1-18. doi : 10.1002/9781118989166.ch1.

[58] D. Fleisch, « A student's guide to Maxwell's equations », in Cambridge, UK : Cambridge University Press, 2008.

[59] S. Mahajan, « A student's guide to Newton's laws of motion », in Cambridge, UK : Cambridge University Press, 2020.

[60] L. Ljung, « Black-box models from input-output measurements », in *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No.01CH 37188)*, t. 1, 2001, 138-146 vol.1. doi : 10.1109/IMTC.2001.928802.

[61] E. T. Maddalena, Y. Lian et C. N. Jones, « Data-driven methods for building control — A review and promising future directions », *Control Engineering Practice*, **95**, p. 104 211, 2020. doi : 10.1016/j.conengprac.2019.104211.

[62] J. K. Huusom, N. K. Poulsen, S. B. Jørgensen et J. B. Jørgensen, « Tuning of methods for offset free MPC based on ARX model representations », in *Proceedings of the 2010 American Control Conference*, 2010, p. 2355-2360. doi : 10.1109/ACC.2010.5530560.

[63] Y. LeCun, Y. Bengio et G. Hinton, « Deep learning », *Nature*, **521**, *7553*, p. 436-444, mai 2015. doi : 10.1038/nature14539.

[64] W. C. Wong, E. Chee, J. Li et X. Wang, « Recurrent neural network-based model predictive control for continuous pharmaceutical manufacturing », *Mathematics*, **6**, *11*, 2018. doi : 10.3390/math6110242.

[65] Z. Wu et P. D. Christofides, « Economic machine-learning-based predictive control of nonlinear systems », *Mathematics*, **7**, *6*, 2019. doi : 10.3390/math7060494.

[66] E. Terzi, T. Bonetti, D. Saccani, M. Farina, L. Fagiano et R. Scattolini, « Learning-based predictive control of the cooling system of a large business centre », *Control Engineering Practice*, **97**, p. 104 348, 2020. doi : 10.1016/j.conengprac.2020.104348.

[67] M. W. Ahmad, A. Mouraud, Y. Rezgui et M. Mourshed, « Deep highway networks and tree-based ensemble for predicting short-term building energy consumption », *Energies*, **11**, *12*, 2018. doi : 10.3390/en11123408.

[68] S. M. Siniscalchi, T. Svendsen et C.-H. Lee, « An artificial neural network approach to automatic speech processing », *Neurocomputing*, **140**, p. 326-338, 2014. doi : 10.1016/j.neucom.2014.03.005.

[69] A. Jain, R. Duin et J. Mao, « Statistical pattern recognition : a review », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, *1*, p. 4-37, 2000. doi : 10.1109/34.824819.

[70] R. Selmic et F. Lewis, « Neural-network approximation of piecewise continuous functions : application to friction compensation », *IEEE Transactions on Neural Networks*, **13**, *3*, p. 745-751, 2002. doi : 10.1109/TNN.2002.1000141.

[71] J. Kuschewski, S. Hui et S. Zak, « Application of feedforward neural networks to dynamical system identification and control », *IEEE Transactions on Control Systems Technology*, **1**, *1*, p. 37-49, 1993. doi : 10.1109/87.221350.

[72] P. Loiseau, C. N. E. Boultifat, P. Chevrel, F. Claveau, S. Espié et F. Mars, « Rider model identification : neural networks and quasi-LPV models », *IET Intelligent Transport Systems*, **14**, *10*, p. 1259-1264, 2020. doi : 10.1049/iet-its.2020.0088.

[73] Y. Wang, « A new concept using LSTM Neural Networks for dynamic system identification », in *2017 American Control Conference (ACC)*, 2017, p. 5324-5329. doi : 10.23919/ACC.2017.7963782.

[74] A. Rehmer et A. Kroll, « On using gated recurrent units for nonlinear system identification », in *2019 18th European Control Conference (ECC)*, 2019, p. 2504-2509. doi : 10.23919/ECC.2019.8795631.

[75] N. Lanzetti, Y. Z. Lian, A. Cortinovis, L. Dominguez, M. Mercangöz et C. Jones, « Recurrent neural network based MPC for process industries », in *2019 18th European Control Conference (ECC)*, 2019, p. 1005-1010. doi : 10.23919/ECC.2019.8795809.

[76] T. Krause, G. Andersson, K. Fröhlich et A. Vaccaro, « Multiple-energy carriers : Modeling of production, delivery, and consumption », *Proceedings of the IEEE*, **99**, *1*, p. 15-27, 2011. doi : 10.1109/JPROC.2010.2083610.

[77] P. Favre-Perrod, « A vision of future energy networks », in *2005 IEEE Power Engineering Society Inaugural Conference and Exposition in Africa*, 2005, p. 13-17. doi : 10.1109/PESAFR.2005.1611778.

[78] M. Mohammadi, Y. Noorollahi, B. Mohammadi-ivatloo et H. Yousefi, « Energy hub : From a model to a concept – A review », *Renewable and Sustainable Energy Reviews*, **80**, p. 1512-1527, 2017. doi : 10.1016/j.rser.2017.07.030.

[79] M. Geidl, G. Koeppel, P. Favre-Perrod, B. Klockl, G. Andersson et K. Frohlich, « Energy hubs for the future », *IEEE Power and Energy Magazine*, **5**, *1*, p. 24-30, 2007. doi : 10.1109/MPAE.2007.264850.

[80] H. Sadeghi, M. Rashidinejad, M. Moeini-Aghtaie et A. Abdollahi, « The energy hub : An extensive survey on the state-of-the-art », *Applied Thermal Engineering*, **161**, p. 114 071, 2019. doi : 10.1016/j.applthermaleng.2019.114071.

[81] G. T. Ayele, « Exploiting the synergies from coupled electricity and heat distribution networks : modelling, simulation and optimization based on an extended energy hub approach », thèse de doct., IMT Atlantique, 2020. adresse : hal:tel-0313267.

[82] M. Mohammadi, Y. Noorollahi, B. Mohammadi-ivatloo, H. Yousefi et S. Jalilinasrabady, « Optimal scheduling of energy hubs in the presence of uncertainty-a review », *Journal of Energy Management and Technology*, **1**, *1*, p. 1-17, 2017. doi : 10.22109/jemt.2017.49432.

[83] P. C. Blaud, P. Haurant, F. Claveau, B. Lacarrière, P. Chevrel et A. Mouraud, « Modelling and control of multi-energy systems through multi-prosumer node and economic model predictive control », *International Journal of Electrical Power & Energy Systems*, **118**, p. 105 778, 2020. doi : 10.1016/j.ijepes.2019.105778.

[84] S. R. Hirshorn, L. D. Voss et L. K. Bromley, *Nasa systems engineering handbook*. 2017. adresse : https://www.nasa.gov/seh/index.html.

[85] P. C. Blaud, P. Chevrel, F. Claveau, P. Haurant et A. Mouraud, « From multi-physics models to neural network for predictive control synthesis », *Optimal Control Applications and Methods*, **n/a**, *n/a*, 2021. doi : 10.1002/oca.2845.

[86] ——, « ResNet and PolyNet based identification and (MPC) control of dynamical systems : a promising way », *n/a*, **n/a**, *n/a*, n/a, n/a.

[87] The MathWorks Inc., *MATLAB and simscape toolbox*, 2021.

[88] S. E. Mattsson, H. Elmqvist et M. Otter, « Physical system modeling with Modelica », *Control Engineering Practice*, **6**, *4*, p. 501-510, 1998. doi : 10.1016/S0967-0661(98)00047-1.

[89] H. Elmqvist et M. Otter, « Innovations for future Modelica », in *12th International Modelica Conference*, sér. Linköping Electronic Conference Proceedings, Linköping University Electronic Press, 2017.

[90] M. Wetter, W. Zuo, T. S. Nouidui et X. Pang, « Modelica buildings library », *Journal of Building Performance Simulation*, **7**, *4*, p. 253-270, 2014. doi : 10.1080/19401493.2013.765506.

[91] G. Plessis, A. Kaemmerlen et A. Lindsay, « BuildSysPro : a Modelica library for modelling buildings and energy systems », in *Proceedings of the 10th International Modelica Conference*, t. 96, Lund, Sweden, 2014, p. 1161-1169. doi : 10.3384/ecp140961161.

[92] Q. Altes-Buch, S. Quoilin et V. Lemort, « Greenhouses : A Modelica library for the simulation of greenhouse climate and energy systems », in *Proceedings of the 13th International Modelica Conference*, t. 157, Regensburg, Germany, 2019, p. 533-542. doi : 10.3384/ecp19157533.

[93] L. Giraud, R. Baviere, M. Vallée et C. Paulus, « Presentation, validation and application of the District Heating Modelica library », in *Proceedings of the 11th International Modelica Conference*, t. 118, Versailles, France, 2015, p. 79-88. doi : 10.3384/ecp1511879.

[94] L. Andresen, P. Dubucq, R. P. Garcia, G. Ackermann, A. Kather et G. Schmitz, « Status of the TransiEnt library : Transient simulation of coupled energy networks with high share of renewable energy », in *Proceedings of the 11th International Modelica Conference*, t. 118, Versailles, France, 2015, p. 695-705. doi : 10.3384/ecp15118695.

[95] M. Dempsey, « Dymola for multi-engineering modelling and simulation », in *2006 IEEE Vehicle Power and Propulsion Conference*, 2006, p. 1-6. doi : 10.1109/VPPC.2006.364294.

[96] P. Fritzson, A. Pop, A. Asghar et al., « The OpenModelica integrated modeling, simulation and optimization environment », in *Proceedings of The American Modelica Conference 2018*, t. 154, Cambridge MA, USA, 2018, p. 206-219. doi : 10.3384/ECP18154206.

[97] L. R. Petzold, « Description of dassl : a differential/algebraic system solver », Sandia National Laboratories, Livermore, CA, USA, rapp. tech., 1982.

[98] L. Liu, F. Felgner et G. Frey, « Comparison of 4 numerical solvers for stiff and hybrid systems simulation », in *2010 IEEE 15th Conference on Emerging Technologies Factory Automation (ETFA 2010)*, 2010, p. 1-8. doi : 10.1109/ETFA.2010.5641330.

[99] P. Fritzson, A. Pop, M. Sjölund et A. Asghar, « MetaModelica – a symbolic-numeric modelica language and comparison to julia », in *Proceedings of the 13th International Modelica Conference*, t. 157, Regensburg, Germany, 2019, p. 289-302. doi : 10.3384/ecp19157289.

[100] A. Pop, P. Östlund, F. Casella, M. Sjölund et R. Franke, « A new openModelica compiler high performance frontend », in *Proceedings of the 13th International Modelica Conference*, t. 157, Regensburg, Germany, 2019, p. 689-698. doi : 10.3384/ecp19157289.

[101] J. Tinnerholm, M. Sjölund et A. Pop, « Towards introducing just-in-time compilation in a Modelica compiler », in *Proceedings of the 9th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools*, sér. EOOLT '19, Berlin, Germany : Association for Computing Machinery, 2019, p. 11-19. doi : 10.1145/3365984.3365990.

[102] W. Xu, « The design and implementation of the μModelica compiler », mém. de mast., Université McGill, Qc, Canada, 2005. adresse : https://escholarship.mcgill.ca/concern/theses/q524jp249.

[103] A. Benveniste, B. Caillaud et M. Malandain, « The mathematical foundations of physical systems modeling languages », *Annual Reviews in Control*, **50**, p. 72-118, 2020. doi : 10.1016/j.arcontrol.2020.08.001.

[104] « System identification : The bases », in *Digital control systems : Design, identification and implementation*. London : Springer London, 2006, p. 201-245. doi : 10.1007/978-1-84628-056-6_5.

[105] F. Bonassi, E. Terzi, M. Farina et R. Scattolini, « LSTM neural networks : Input to state stability and probabilistic safety verification », in *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, A. M. Bayen, A. Jadbabaie, G. Pappas et al., éd., sér. Proceedings of Machine Learning Research, t. 120, PMLR, 2020, p. 85-94.

[106] P. Potočnik, B. Vidrih, A. Kitanovski et E. Govekar, « Neural network, ARX, and extreme learning machine models for the short-term prediction of temperature in buildings », *Building Simulation*, **12**, p. 1077-1093, 2019. doi : 10.1007/s12273-019-0548-y5.

[107] W. Gerstner, W. M. Kistler, R. Naud et L. Paninski, *Neuronal dynamics : From single neurons to networks and models of cognition*. Cambridge, UK : Cambridge University Press, 2014.

[108] LadyofHats, Wikimedia Commons. adresse : https://fr.wikipedia.org/wiki/Neurone#/media/Fichier:Complete_neuron_cell_diagram_fr.svg.

[109] X. S. Yang, *Nature-inspired metaheuristic algorithms*. Frome, UK : Luniver Press, 2010.

[110] J. Y. Lettvin, H. R. Maturana, W. S. McCulloch et W. H. Pitts, « What the frog's eye tells the frog's brain », *Proceedings of the IRE*, **47**, p. 1940-1951, 1959. doi : 10.1109/JRPROC.1959.287207.

[111] S. Chakraverty, D. M. Sahoo et N. R. Mahato, « McCulloch–Pitts neural network model », in *Concepts of soft computing : Fuzzy and ANN with programming*. Singapore : Springer Singapore, 2019, p. 167-173. doi : 10.1007/978-981-13-7430-2_11.

[112] F. Li F., R. Krishna, D. Xu et al., *Convolutional neural networks for visual recognition*, CS231n, Stanford School of Engineering, Stanford University, 2021. adresse : http://cs231n.stanford.edu/.

[113] M. T. Hagan, H. B. Demuth et M. Beale, *Neural network design*. 1997.

[114] A. D. Rasamoelina, F. Adjailia et P. Sinčák, « A Review of activation function for artificial neural network », in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2020, p. 281-286. doi : 10.1109/SAMI48414.2020.9108717.

[115] G. Cybenko, « Approximation by superpositions of a sigmoidal function », *Mathematics of Control, Signals and Systems*, **2**, *4*, p. 303-314, déc. 1989. doi : 10.1007/BF02551274.

[116] V. K. Ojha, A. Abraham et V. Snášel, « Metaheuristic design of feedforward neural networks : A review of two decades of research », *Engineering Applications of Artificial Intelligence*, **60**, p. 97-116, 2017. doi : 10.1016/j.engappai.2017.01.013.

[117] K. Hornik, « Approximation capabilities of multilayer feedforward networks », *Neural Networks*, **4**, *2*, p. 251-257, 1991. doi : 10.1016/0893-6080(91)90009-T.

[118] K.-K. K. Kim, E. R. Patrón et R. D. Braatz, « Standard representation and unified stability analysis for dynamic artificial neural network models », *Neural Networks*, **98**, p. 251-262, 2018. doi : 10.1016/j.neunet.2017.11.014.

[119] K. He, X. Zhang, S. Ren et J. Sun, « Deep residual learning for image recognition », in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[120] ——, « Identity mappings in deep residual networks », in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe et M. Welling, éd., Cham : Springer International Publishing, 2016, p. 630-645.

[121] D. Silver, J. Schrittwieser, K. Simonyan et al., « Mastering the game of Go without human knowledge », *Nature*, **550**, *7676*, p. 354-359, oct. 2017. doi : 10.1038/nature24270.

[122] A. Vaswani, N. Shazeer, N. Parmar et al., « Attention is all you need », in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, sér. NIPS'17, Long Beach, California, USA : Curran Associates Inc., 2017, p. 6000-6010.

[123] S. Ouala, A. Pascual et R. Fablet, « Residual integration neural network », in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, p. 3622-3626. doi : 10.1109/ICASSP.2019.8683447.

[124] E. Weinan, « A proposal on machine learning via dynamical systems », *Communications in Mathematics and Statistics*, **5**, *1*, p. 1-11, mars 2017. doi : 10.1007/s40304-017-0103-z.

[125] E. Hairer, G. Wanner et S. P. Nørsett, « Classical Mathematical Theory », in *Solving ordinary differential equations I : Nonstiff problems*. Berlin, Heidelberg : Springer Berlin Heidelberg, 1993, p. 1-128. doi : 10.1007/978-3-540-78862-1_1.

[126] X. Zhang, Z. Li, C. Change Loy et D. Lin, « PolyNet : A pursuit of structural diversity in very deep networks », in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[127] Y. Lu, A. Zhong, Q. Li et B. Dong, « Beyond finite layer neural networks : bridging deep architectures and numerical differential equations », in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy et A. Krause, éd., sér. Proceedings of Machine Learning Research, t. 80, PMLR, 2018, p. 3276-3285.

[128] Y. Lecun, « Une procedure d'apprentissage pour reseau a seuil asymmetrique (A learning scheme for asymmetric threshold networks) », *Proceedings of Cognitiva 85*, p. 599-604, 1985.

[129] D. E. Rumelhart, G. E. Hinton et R. J. Williams, « Learning representations by back-propagating errors », *Nature*, **323**, *6088*, p. 533-536, oct. 1986. doi : 10.1038/323533a0.

[130] J. Zhang et I. Mitliagkas, « YellowFin and the art of momentum tuning », in *Proceedings of machine learning and systems*, A. Talwalkar, V. Smith et M. Zaharia, éd., t. 1, 2019, p. 289-308.

[131] L. Yang et A. Shami, « On hyperparameter optimization of machine learning algorithms : Theory and practice », *Neurocomputing*, **415**, p. 295-316, 2020. doi : 10.1016/j.neucom.2020.07.061.

[132] M. Abdel-Basset, L. Abdel-Fatah et A. K. Sangaiah, « Chapter 10 - Metaheuristic algorithms : A comprehensive review », in *Computational intelligence for multimedia big data on the cloud with engineering applications*, sér. Intelligent Data-Centric Systems, A. K. Sangaiah, M. Sheng et Z. Zhang, éd., Academic Press, 2018, p. 185-231. doi : 10.1016/B978-0-12-813314-9.00010-4.

[133] R. Feldt et A. Stukalov, *BlackBoxOptim.jl*, GitHub repository, 2018. adresse : https://github.com/robertfeldt/BlackBoxOptim.jl.

[134] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau et C. Gagné, « DEAP : Evolutionary algorithms made easy », *Journal of Machine Learning Research*, **13**, *70*, p. 2171-2175, 2012. adresse : http://jmlr.org/papers/v13/fortin12a.html.

[135] Z. Wang et A. C. Bovik, « Mean squared error : Love it or leave it ? A new look at Signal Fidelity Measures », *IEEE Signal Processing Magazine*, **26**, *1*, p. 98-117, 2009. doi : 10.1109/MSP.2008.930649.

[136] M. Innes, « Flux : Elegant machine learning with Julia », *Journal of Open Source Software*, **3**, *25*, p. 602, 2018. doi : 10.21105/joss.00602.

[137] D. Yuret, « Knet : beginning deep learning with 100 lines of julia », in *30th Conference on Neural Information Processing Systems (NIPS 2016)*, 2016.

[138] The MathWorks Inc., *MATLAB and machine learning toolbox*, 2021.

[139] N. Ketkar, « Introduction to PyTorch », in *Deep learning with Python : A hands-on introduction*. Berkeley, CA : Apress, 2017, p. 195-208. doi : 10.1007/978-1-4842-2766-4_12.

[140] J. Moolayil, « An introduction to deep learning and Keras », in *Learn Keras for deep neural networks : A fast-track approach to modern deep learning with Python*. Berkeley, CA : Apress, 2019, p. 1-16. doi : 10.1007/978-1-4842-4240-7_1.

[141] T. Besard, C. Foket et B. De Sutter, « Effective extensible programming : unleashing Julia on GPUs », *IEEE Transactions on Parallel and Distributed Systems*, **30**, *4*, p. 827-841, 2019. doi : 10.1109/TPDS.2018.2872064.

[142] N. Jouppi, C. Young, N. Patil et D. Patterson, « Motivation for and evaluation of the first tensor processing unit », *IEEE Micro*, **38**, *3*, p. 10-19, 2018. doi : 10.1109/MM.2018.032271057.

[143] M. Zeilinger, *Learning-based predictive control*, IGSC2021-M01, International Graduate School on Control, European Embedded Control Institute, 2021.

[144] J. B. Rawlings, D. Q. Mayne et M. Diehl, *Model predictive control : theory, computation, and design*. Madison : Princeton University Press, 2017.

[145] R. Bhatia, *Positive definite matrices*. New Jersey : Princeton University Press, 2015.

[146] D. Mayne, J. Rawlings, C. Rao et P. Scokaert, « Constrained model predictive control : Stability and optimality », *Automatica*, **36**, *6*, p. 789-814, 2000. doi : 10.1016/S0005-1098(99)00214-9.

[147] H. CHEN et F. ALLGöWER, « A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability », *Automatica*, **34**, *10*, p. 1205-1217, 1998. doi : 10.1016/S0005-1098(98)00073-9.

[148] M. Murillo, G. Sánchez et L. Giovanini, « Iterated non-linear model predictive control based on tubes and contractive constraints », *ISA Transactions*, **62**, p. 120-128, 2016. doi : 10.1016/j.isatra.2016.01.008.

[149] B. O'Donoghue, E. Chu, N. Parikh et S. Boyd, « Conic optimization via operator splitting and homogeneous self-dual embedding », *Journal of Optimization Theory and Applications*, **169**, *3*, p. 1042-1068, 2016. doi : 10.1007/s10957-016-0892-3.

[150] J. A. Rossiter, *Model-based predictive control : a practical approach*. Boca Raton : CRC Press, 2003. doi : 10.1201/9781315272610.

[151] P. Scokaert, D. Mayne et J. Rawlings, « Suboptimal model predictive control (feasibility implies stability) », *IEEE Transactions on Automatic Control*, **44**, *3*, p. 648-654, 1999. doi : 10.1109/9.751369.

[152] S. de Oliveira Kothare et M. Morari, « Contractive model predictive control for constrained nonlinear systems », *IEEE Transactions on Automatic Control*, **45**, *6*, p. 1053-1071, 2000. doi : 10.1109/9.863592.

[153] F. Borrelli, A. Bemporad et M. Morari, *Predictive control for linear and hybrid systems*. Cambridge, UK : Cambridge University Press, 2017.

[154] B. Kouvaritakis et M. Cannon, *Model predictive control*. Springer International Publishing, 2016. doi : 10.1007/978-3-319-24853-0.

[155] E. Kerrigan et J. Maciejowski, « Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control », in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*, t. 5, 2000, 4951-4956 vol.5. doi : 10.1109/CDC.2001.914717.

[156] V. T. Minh et N. Afzulpurkar, « A comparative study on computational schemes for nonlinear model predictive control », *Asian Journal of Control*, **8**, *4*, p. 324-331, 2006. doi : 10.1111/j.1934-6093.2006.tb00284.x.

[157] D. M. Gay, « The AMPL modeling language : An aid to formulating and solving optimization problems », in *Numerical analysis and optimization*, M. Al-Baali, L. Grandinetti et A. Purnama, éd., Cham : Springer International Publishing, 2015, p. 95-116. doi : 10.1007/978-3-319-17689-5_5..

[158] J. Bisschop et A. Meeraus, « On the development of a general algebraic modeling system in a strategic planning environment », in *Applications*, J.-L. Goffin et J.-M. Rousseau, éd. Berlin, Heidelberg : Springer Berlin Heidelberg, 1982, p. 1-29. doi : 10.1007/BFb0121223.

[159] The MathWorks Inc., *MATLAB and optimization toolbox*, 2021.

[160] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings et M. Diehl, « CasADi : a software framework for nonlinear optimization and optimal control », *Mathematical Programming Computation*, **11**, *1*, p. 1-36, 2019. doi : 10.1007/s12532-018-0139-4.

[161] J. Lofberg, « YALMIP : a toolbox for modeling and optimization in MATLAB », in *2004 IEEE International Conference on Robotics and Automation*, 2004, p. 284-289. doi : 10.1109/CACSD.2004.1393890.

[162] W. E. Hart, C. D. Laird, J.-P. Watson et al., « Introduction », in *Pyomo — optimization modeling in Python*. Cham : Springer International Publishing, 2017, p. 1-11. doi : 10.1007/978-3-319-58821-6_1.

[163] L. Perron, « Operations research and constraint programming at Google », in *Principles and practice of constraint programming – CP 2011*, J. Lee, éd., Berlin, Heidelberg : Springer Berlin Heidelberg, 2011, p. 2-2.

[164] I. Dunning, J. Huchette et M. Lubin, « JuMP : A modeling language for mathematical optimization », *SIAM review*, **59**, *2*, p. 295-320, 2017. doi : 10.1137/15M1020575.

[165] J. Bezanson, A. Edelman, S. Karpinski et V. B. Shah, « Julia : A fresh approach to numerical computing », *SIAM review*, **59**, *1*, p. 65-98, 2017. doi : 10.1137/141000671.

[166] M. Diehl, H. J. Ferreau et N. Haverbeke, « Efficient numerical methods for nonlinear MPC and moving horizon estimation », in *Nonlinear model predictive control : Towards new challenging applications*, L. Magni, D. M. Raimondo et F. Allgöwer, éd. Berlin, Heidelberg : Springer Berlin Heidelberg, 2009, p. 391-417. doi : 10.1007/978-3-642-01094-1_32.

[167] L. Grüne et J. Pannek, « Numerical optimal control of nonlinear systems », in *Nonlinear model predictive control : theory and algorithms*. Cham : Springer International Publishing, 2017, p. 367-434. doi : 10.1007/978-3-319-46024-6_12.

[168] E. M. Gertz et S. J. Wright, « Object-oriented software for quadratic programming », *ACM Transactions on Mathematical Software*, **29**, *1*, p. 58-81, mars 2003. doi : 10.1145/641876.641880.

[169] B. Stellato, G. Banjac, P. Goulart, A. Bemporad et S. Boyd, « OSQP : an operator splitting solver for quadratic programs », *Mathematical Programming Computation*, **12**, *4*, p. 637-672, 2020. doi : 10.1007/s12532-020-00179-2.

[170] M. Garstka, M. Cannon et P. Goulart, « COSMO : A conic operator splitting method for large convex problems », in *2019 18th European Control Conference (ECC)*, 2019, p. 1951-1956. doi : 10.23919/ECC.2019.8796161.

[171] A. Wächter et L. T. Biegler, « On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming », *Mathematical Programming*, **106**, *1*, p. 25-57, 2006. doi : 10.1007/s10107-004-0559-y.

[172] R. H. Byrd, J. Nocedal et R. A. Waltz, « Knitro : An integrated package for nonlinear optimization », in *Large-scale nonlinear optimization*, G. Di Pillo et M. Roma, éd. Boston, MA : Springer US, 2006, p. 35-59. doi : 10.1007/0-387-30065-1_4.

[173] N. Karmarkar, « A new polynomial-time algorithm for linear programming », in *STOC '84*, New York, NY, USA : Association for Computing Machinery, 1984, p. 302-311. doi : 10.1145/800057.808695.

[174] R. H. Byrd, M. E. Hribar et J. Nocedal, « An interior point algorithm for large-scale nonlinear programming », *SIAM Journal on Optimization*, **9**, *4*, p. 877-900, 1999. doi : 10.1137/S1052623497325107.

[175] S. Boyd, N. Parikh, E. Chu, B. Peleato et J. Eckstein. Now Publishers Inc., 2011. doi : 10.1561/2200000016.

[176] M. Schubiger, G. Banjac et J. Lygeros, « GPU acceleration of ADMM for large-scale quadratic programming », *Journal of Parallel and Distributed Computing*, **144**, p. 55-67, 2020. doi : 10.1016/j.jpdc.2020.05.021.

[177] K. M. Abughalieh et S. G. Alawneh, « A survey of parallel implementations for model predictive control », *IEEE Access*, **7**, p. 34 348-34 360, 2019. doi : 10.1109/ACCESS.2019.2904240.

[178] D. Bensoussan, *Commande moderne : approche par modèles continus et discrets*. Montréal : Presses Internationales Polytechnique, 2008.

[179] J.-J. Wang, « Simulation studies of inverted pendulum based on PID controllers », *Simulation Modelling Practice and Theory*, **19**, *1*, p. 440-449, 2011, Modeling and Performance Analysis of Networking and Collaborative Systems, issn : 1569-190X. doi : 10.1016/j.simpat.2010.08.003.

[180] G. A. Hicks et W. H. Ray, « Approximation methods for optimal control synthesis », *The Canadian Journal of Chemical Engineering*, **49**, *4*, p. 522-528, 1971. doi : 10.1002/cjce.5450490416.

[181] M. Ellis et P. D. Christofides, « Integrating dynamic economic optimization and model predictive control for optimal operation of nonlinear process systems », *Control Engineering Practice*, **22**, p. 242-251, 2014. doi : 10.1016/j.conengprac.2013.02.016.

[182] D. Q. Mayne, E. C. Kerrigan, E. J. van Wyk et P. Falugi, « Tube-based robust nonlinear model predictive control », *International Journal of Robust and Nonlinear Control*, **21**, *11*, p. 1341-1353, 2011. doi : 10.1002/rnc.1758.

[183] M. Hertneck, J. Köhler, S. Trimpe et F. Allgöwer, « Learning an approximate model predictive controller with guarantees », *IEEE Control Systems Letters*, **2**, *3*, p. 543-548, 2018. doi : 10.1109/LCSYS.2018.2843682.

[184] K. Johansson, « The quadruple-tank process : a multivariable laboratory process with an adjustable zero », *IEEE Transactions on Control Systems Technology*, **8**, *3*, p. 456-465, 2000. doi : 10.1109/87.845876.

[185] K. Johansson et J. Nunes, « A multivariable laboratory process with an adjustable zero », in *Proceedings of the 1998 American Control Conference (ACC)*, t. 4, 1998, 2045-2049 vol.4. doi : 10.1109/ACC.1998.702986.

[186] K. Johansson, A. Horch, O. Wijk et A. Hansson, « Teaching multivariable control using the quadruple-tank process », in *Proceedings of the 38th IEEE Conference on Decision and Control (CDC)*, t. 1, 1999, 807-812 vol.1. doi : 10.1109/CDC.1999.832889.

[187] I. Alvarado, D. Limon, D. Muñoz de la Peña et al., « A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark », *Journal of Process Control*, **21**, *5*, p. 800-815, 2011, Special Issue on Hierarchical and Distributed Model Predictive Control. doi : 10.1016/j.jprocont.2011.03.003.

[188] S. B. Prusty, S. Seshagiri, U. C. Pati et K. K. Mahapatra, « Sliding mode control of coupled tank systems using conditional integrators », *IEEE/CAA Journal of Automatica Sinica*, **7**, *1*, p. 118-125, 2020. doi : 10.1109/JAS.2019.1911831.

[189] H. Pan, H. Wong, V. Kapila et M. S. de Queiroz, « Experimental validation of a nonlinear backstepping liquid level controller for a state coupled two tank system », *Control Engineering Practice*, **13**, *1*, p. 27-40, 2005. doi : 10.1016/j.conengprac.2003.12.019.

[190] G. Balasubramanian, K. Hariprasad, N. Sivakumaran et T. K. Radhakrishnan, « Adaptive control of multivariable process using recurrent neural networks », *Instrumentation Science & Technology*, **37**, *6*, p. 615-630, 2009. doi : 10.1080/10739140903251933.

[191] B. S. Sousa, F. V. Silva et A. M. F. Fileti, « Level control of coupled tank system based on neural network techniques », *Chemical Product and Process Modeling*, **15**, *3*, p. 20 190 086, 2020. doi : doi:10.1515/cppm-2019-0086.

[192] M. Lazar, « Model predictive control of hybrid systems : stability and robustness », thèse de doct., Technische Universiteit Eindhoven, 2006. doi : 10.6100/IR612103.

[193] J. Drgoňa, D. Picard, M. Kvasnica et L. Helsen, « Approximate model predictive building control via machine learning », *Applied Energy*, **218**, p. 199-216, 2018. doi : 10.1016/j.apenergy.2018.02.156.

[194] D. Picard, F. Jorissen et L. Helsen, « Methodology for obtaining linear state space building energy simulation models », in *Proceedings of the 11th International Modelica Conference*, t. 118, Versailles, France, 2015, p. 51-58. doi : 10.3384/ecp140961161.

[195] D. Smith et H. LeBlanc, « Variable activation functions and spawning in neuroevolution », in *Proceedings of the 2018 ASEE North Central Section Conference*, Akron, OH, USA, 2018.

[196] P. Ramachandran, B. Zoph et V. Q. Le, « Searching for activation functions », in *International Conference on Learning Representations (ICLR)*, 2018.

[197] D. P. Kingma et J. Ba, « Adam : A method for stochastic optimization », in *International Conference on Learning Representations (ICLR)*, 2015.

[198] L. Liu, H. Jiang, P. He et al., « On the variance of the adaptive learning rate and beyond », in *International Conference on Learning Representations (ICLR)*, 2020.

[199] T. Dozat, « Incorporating nesterov momentum into adam », in *International Conference on Learning Representations (ICLR)*, 2016.

[200] C. Daskalakis, A. Ilyas, V. Syrgkanis et H. Zeng, « Training GANs with optimism », in *International Conference on Learning Representations (ICLR)*, 2018.

[201] Y. Sun, S. Vogel et H. Steuer, « Combining advantages of specialized simulation tools and Modelica models using Functional Mock-up Interface (FMI) », in *Proceedings of the 8th International Modelica Conference*, t. 63, Dresden, Germany, 2011, p. 491-494. doi : 10.3384/ecp11063491.

[202] P. C. Blaud, P. Chevrel, F. Claveau, P. Haurant et A. Mouraud, « Multi-flux optimisation of a greenhouse with a model predictive control », *n/a*, **n/a**, *n/a*, n/a, n/a.

[203] RaboResearch, *The 2018 World Vegetable Map*, Rabobank, 2018.

[204] F. Golzar, N. Heeren, S. Hellweg et R. Roshandel, « A novel integrated framework to evaluate greenhouse energy demand and crop yield production », *Renewable and Sustainable Energy Reviews*, **96**, p. 487-501, 2018. doi : 10.1016/j.rser.2018.06.046.

[205] Gestebier, Wikimedia Commons. adresse : https://commons.wikimedia.org/wiki/File:Sierteeltbedrijf.jpg.

[206] Y. E. Afou, « Contribution au contrôle des paramètres climatiques sous serre », thèse de doct., Université des Sciences et Technologies de Lille 1, 2014.

[207] B. Vanthoor, C. Stanghellini, E. van Henten et P. de Visser, « A methodology for model-based greenhouse design : Part 1, a greenhouse climate model for a broad range of designs and climates », *Biosystems Engineering*, **110**, *4*, p. 363-377, 2011. doi : 10.1016/j.biosystemseng.2011.06.001.

[208] L. Chen, S. Du, D. Xu, Y. He et M. Liang, « Sliding mode control based on disturbance observer for greenhouse climate systems », *Mathematical Problems in Engineering*, **2018**, p. 2 071 585, avr. 2018. doi : 10.1155/2018/2071585.

[209] F. Lafont et J.-F. Balmat, « Optimized fuzzy control of a greenhouse », *Fuzzy Sets and Systems*, **128**, *1*, p. 47-59, 2002, ACIDCA. doi : 10.1016/S0165-0114(01)00182-8.

[210] F. Lafont, J.-F. Balmat, N. Pessel et M. Fliess, « A model-free control strategy for an experimental greenhouse with an application to fault accommodation », *Computers and Electronics in Agriculture*, **110**, p. 139-149, 2015. doi : 10.1016/j.compag.2014.11.008.

[211] E. Van Henten et J. Bontsema, « Time-scale decomposition of an optimal control problem in greenhouse climate management », *Control Engineering Practice*, **17**, *1*, p. 88-96, 2009. doi : 10.1016/j.conengprac.2008.05.008.

[212] D. Xu, S. Du et G. van Willigenburg, « Adaptive two time-scale receding horizon optimal control for greenhouse lettuce cultivation », *Computers and Electronics in Agriculture*, **146**, p. 93-103, 2018. doi : 10.1016/j.compag.2018.02.001.

[213] K. Kodra et Z. Gajic, « Optimal control for a new class of singularly perturbed linear systems », *Automatica*, **81**, p. 203-208, 2017. doi : 10.1016/j.automatica.2017.03.017.

[214] A. Ramírez-Arias, F. Rodríguez, J. Guzmán et M. Berenguel, « Multiobjective hierarchical control architecture for greenhouse crop growth », *Automatica*, **48**, *3*, p. 490-498, 2012, issn : 0005-1098. doi : 10.1016/j.automatica.2012.01.002.

[215] M. C. Bozchalui, C. A. Cañizares et K. Bhattacharya, « Optimal energy management of greenhouses in smart grids », *IEEE Transactions on Smart Grid*, **6**, *2*, p. 827-835, 2015. doi : 10.1109/TSG.2014.2372812.

[216] F. Rodríguez, M. Berenguel et M. R. Arahal, « A hierarchical control system for maximizing profit in greenhouse crop production », in *2003 European Control Conference (ECC)*, 2003, p. 2753-2758. doi : 10.23919/ECC.2003.7086458.

[217] P. van Beveren, J. Bontsema, G. van Straten et E. van Henten, « Optimal control of greenhouse climate using minimal energy and grower defined bounds », *Applied Energy*, **159**, p. 509-519, 2015. doi : 10.1016/j.apenergy.2015.09.012.

[218] P. Van Beveren, J. Bontsema, G. Van Straten et E. Van Henten, « Minimal heating and cooling in a modern rose greenhouse », *Applied Energy*, **137**, p. 97-109, 2015, issn : 0306-2619. doi : 10.1016/j.apenergy.2014.09.083.

[219] I. Ioslovich, P.-O. Gutman et R. Linker, « Hamilton–Jacobi–Bellman formalism for optimal climate control of greenhouse crop », *Automatica*, **45**, *5*, p. 1227-1231, 2009. doi : 10.1016/j.automatica.2008.12.024.

[220] S. Piñón, E. Camacho, B. Kuchen et M. Peña, « Constrained predictive control of a greenhouse », *Computers and Electronics in Agriculture*, **49**, *3*, p. 317-329, 2005, Modelling and control in agricultural processes, issn : 0168-1699. doi : 10.1016/j.compag.2005.08.007.

[221] M. El Ghoumari, H.-J. Tantau et J. Serrano, « Non-linear constrained MPC : Real-time implementation of greenhouse air temperature control », *Computers and Electronics in Agriculture*, **49**, *3*, p. 345-356, 2005, Modelling and control in agricultural processes, issn : 0168-1699. doi : 10.1016/j.compag.2005.08.005.

[222] L. Chen, S. Du, Y. He, M. Liang et D. Xu, « Robust model predictive control for greenhouse temperature based on particle swarm optimization », *Information Processing in Agriculture*, **5**, *3*, p. 329-338, 2018. doi : 10.1016/j.inpa.2018.04.003.

[223] J. Ramos-Teodoro, A. Giménez-Miralles, F. Rodríguez et M. Berenguel, « A flexible tool for modeling and optimal dispatch of resources in agri-energy hubs », *Sustainability*, **12**, *21*, 2020. doi : 10.3390/su12218820.

[224] A. Ouammi, Y. Achour, H. Dagdougui et D. Zejli, « Optimal operation scheduling for a smart greenhouse integrated microgrid », *Energy for Sustainable Development*, **58**, p. 129-137, 2020. doi : 10.1016/j.esd.2020.08.001.

[225] A. Ouammi, Y. Achour, D. Zejli et H. Dagdougui, « Supervisory model predictive control for optimal energy management of networked smart greenhouses integrated microgrid », *IEEE Transactions on Automation Science and Engineering*, **17**, *1*, p. 117-128, 2020. doi : 10.1109/TASE.2019.2910756.

[226] P. van Beveren, J. Bontsema, G. van Straten et E. van Henten, « Optimal utilization of a boiler, combined heat and power installation, and heat buffers in horticultural greenhouses », *Computers and Electronics in Agriculture*, **162**, p. 1035-1048, 2019. doi : 10.1016/j.compag.2019.05.040.

[227] P. van Beveren, J. Bontsema, A. van 't Ooster, G. van Straten et E. van Henten, « Optimal utilization of energy equipment in a semi-closed greenhouse », *Computers and Electronics in Agriculture*, **179**, p. 105 800, 2020, issn : 0168-1699. doi : 10.1016/j.compag.2020.105800.

[228] L. Graamans, A. van den Dobbelsteen, E. Meinen et C. Stanghellini, « Plant factories ; crop transpiration and energy balance », *Agricultural Systems*, **153**, p. 138-147, 2017. doi : 10.1016/j.agsy.2017.01.003.

[229] A. A. Farooq, A. Afram, N. Schulz et F. Janabi-Sharifi, « Grey-box modeling of a low pressure electric boiler for domestic hot water system », *Applied Thermal Engineering*, **84**, p. 257-267, 2015. doi : 10.1016/j.applthermaleng.2015.03.050.

[230] B. J. Huang et P. Y. Ko, « A system dynamics model of fire-tube shell boiler », *Journal of Dynamic Systems, Measurement, and Control*, **116**, *4*, p. 745-754, déc. 1994. doi : 10.1115/1.2899274.

[231] F. Gottelt, T. Hoppe et L. Nielsen, « Applying the power plant library clara for control optimisation », in *Proceedings of the 12th International Modelica Conference*, t. 132, Prague, Czech Republic, 2017, p. 867-877. doi : 10.3384/ecp17132867.

[232] A. Parisio, C. Del Vecchio et A. Vaccaro, « A robust optimization approach to energy hub management », *International Journal of Electrical Power & Energy Systems*, **42**, *1*, p. 98-104, 2012. doi : 10.1016/j.ijepes.2012.03.015.

[233] *Global Modeling and Assimilation Office (GMAO), MERRA-2 tavg1_2d_slv_Nx :2d,1-Hourly,Time-Averaged,Single-Level,Assimilation,Single-Level Diagnostics V5.12.4*, Goddard Earth Sciences Data and Information Services Center (GES DISC), 2015. doi : 10.5067/VJAFPLI1CSIV, disponible. adresse : http://www.soda-pro.com/.

[234] R. Lougee-Heimer, « The common optimization interface for operations research : Promoting open-source software in the operations research community », *IBM Journal of Research and Development*, **47**, *1*, p. 57-66, 2003. doi : 10.1147/rd.471.0057.

[235] Y. Gaoua, « Modèles mathématiques et techniques d'optimisation non linéaire et combinatoire pour la gestion d'énergie d'un système multi-source : vers une implantation temps-réel pour différentes structures électriques de véhicules hybrides », thèse de doct., Institut National Polytechnique de Toulouse, 2014. adresse : hal:tel-01104983.

[236] K. Gao, G. Mei, F. Piccialli, S. Cuomo, J. Tu et Z. Huo, « Julia language in machine learning : Algorithms, applications, and open issues », *Computer Science Review*, **37**, p. 100 254, 2020. doi : 10.1016/j.cosrev.2020.100254.

[237] Dymola, *Dynamic modeling laboratory user's manuel volume 1*, 2021.

[238] A. Bernacchia et S. Pigolotti, « Self-consistent method for density estimation », *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, **73**, *3*, p. 407-422, 2011. doi : 10.1111/j.1467-9868.2011.00772.x.

[239] S. Byrne, *KernelDensity.jl*, GitHub repository, 2014. adresse : https://github.com/JuliaStats/KernelDensity.jl.

[240] Modelica Association, *Modelica – A Unified Object-Oriented Language for Systems Modeling*, GitHub repository, 2021. adresse : https://github.com/modelica/ModelicaSpecification.

[241] Z. Ding, Y. Huang, H. Yuan et H. Dong, « Introduction to reinforcement learning », in *Deep reinforcement learning : Fundamentals, research and applications*, H. Dong, Z. Ding et S. Zhang, éd. Singapore : Springer Singapore, 2020, p. 47-123. doi : 10.1007/978-981-15-4095-0_2.

[242] M. Ellis, H. Durand et P. D. Christofides, « A tutorial review of economic model predictive control methods », *Journal of Process Control*, **24**, *8*, p. 1156-1178, 2014. doi : 10.1016/j.jprocont.2014.03.010.

[243] M. Tanaskovic, L. Fagiano, R. Smith et M. Morari, « Adaptive receding horizon control for constrained MIMO systems », *Automatica*, **50**, *12*, p. 3019-3029, 2014. doi : 10.1016/j.automatica.2014.10.036.

[244] M. Tanaskovic, L. Fagiano et V. Gligorovski, « Adaptive model predictive control for linear time varying MIMO systems », *Automatica*, **105**, p. 237-245, 2019. doi : 10.1016/j.automatica.2019.03.030.

[245] D. A. Copp et J. P. Hespanha, « Simultaneous nonlinear model predictive control and state estimation », *Automatica*, **77**, p. 143-154, 2017. doi : 10.1016/j.automatica.2016.11.041.

[246] S. Lucia, T. Finkler et S. Engell, « Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty », *Journal of Process Control*, **23**, *9*, p. 1306-1319, 2013. doi : 10.1016/j.jprocont.2013.08.008.

[247] D. Q. Mayne, E. C. Kerrigan, E. J. van Wyk et P. Falugi, « Tube-based robust nonlinear model predictive control », *International Journal of Robust and Nonlinear Control*, **21**, *11*, p. 1341-1353, 2011. doi : 10.1002/rnc.1758.

[248] G. Huang, Z. Liu, L. van der Maaten et K. Q. Weinberger, « Densely connected convolutional networks », in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[249] B. Amos, L. Xu et J. Z. Kolter, « Input convex neural networks », in *Proceedings of the 34th International Conference on Machine Learning*, D. Precup et Y. W. Teh, éd., sér. Proceedings of Machine Learning Research, t. 70, PMLR, 2017, p. 146-155.

[250] Y. Chen, Y. Shi et B. Zhang, « Optimal control via neural networks : A convex approach », in *International Conference on Learning Representations*, 2019.

[251] M. Grant, S. Boyd et Y. Ye, « Disciplined convex programming », in *Global optimization : From theory to implementation*, L. Liberti et N. Maculan, éd. Boston, MA : Springer US, 2006, p. 155-210. doi : 10.1007/0-387-30528-9_7.

[252] M. Udell, K. Mohan, D. Zeng, J. Hong, S. Diamond et S. Boyd, « Convex optimization in Julia », in *2014 First Workshop for High Performance Technical Computing in Dynamic Languages*, 2014, p. 18-28. doi : 10.1109/HPTCDL.2014.5.

[253] M. Tan et Q. Le, « EfficientNet : Rethinking model scaling for convolutional neural networks », in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri et R. Salakhutdinov, éd., sér. Proceedings of Machine Learning Research, t. 97, PMLR, 2019, p. 6105-6114.

[254] J. Wang, S. Lu, S.-H. Wang et Y.-D. Zhang, « A review on extreme learning machine », *Multimedia Tools and Applications*, mai 2021. doi : 10.1007/s11042-021-11007-7.

[255] G. Li et J. Shi, « On comparing three artificial neural networks for wind speed forecasting », *Applied Energy*, **87**, *7*, p. 2313-2320, 2010. doi : 10.1016/j.apenergy.2009.12.013.

[256] K. Banerjee, E. Georganas, D. D. Kalamkar et al., « Optimizing deep jearning RNN topologies on intel architecture », *Supercomputing Frontiers and Innovations*, **6**, *3*, p. 64-85, sept. 2019.

[257] M. Schuster et K. Paliwal, « Bidirectional recurrent neural networks », *IEEE Transactions on Signal Processing*, **45**, *11*, p. 2673-2681, 1997. doi : 10.1109/78.650093.

[258] S. Hochreiter et J. Schmidhuber, « Long short-term memory », *Neural Computation*, **9**, *8*, p. 1735-1780, 1997. doi : 10.1162/neco.1997.9.8.1735.

[259] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink et J. Schmidhuber, « LSTM : A search space odyssey », *IEEE Transactions on Neural Networks and Learning Systems*, **28**, *10*, p. 2222-2232, 2017. doi : 10.1109/TNNLS.2016.2582924.

[260] S. Wang, K. Fan, N. Luo et al., « Massive computational acceleration by using neural networks to emulate mechanism-based biological models », *Nature Communications*, **10**, *1*, p. 4354, sept. 2019. doi : 10.1038/s41467-019-12342-y.

[261] R. Dey et F. M. Salem, « Gate-variants of Gated Recurrent Unit (GRU) neural networks », in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2017, p. 1597-1600. doi : 10.1109/MWSCAS.2017.8053243.

[262] M. Ravanelli, P. Brakel, M. Omologo et Y. Bengio, « Light gated recurrent units for speech recognition », *IEEE Transactions on Emerging Topics in Computational Intelligence*, **2**, *2*, p. 92-102, 2018. doi : 10.1109/TETCI.2017.2762739.

[263] Y.-J. Wang et C.-T. Lin, « Runge-Kutta neural network for identification of dynamical systems in high accuracy », *IEEE Transactions on Neural Networks*, **9**, *2*, p. 294-307, 1998. doi : 10.1109/72.661124.

[264] M. Zhu, B. Chang et C. Fu, *Convolutional neural networks combined with Runge-Kutta methods*, 2019. arXiv : 1802.08831 [cs.CV].

[265] G. Larsson, M. Maire et G. Shakhnarovich, « FractalNet : Ultra-deep neural networks without residuals », 2016. arXiv : 1605.07648.

[266] E. Süli et D. F. Mayers, *An introduction to numerical analysis*. Cambridge, UK : Cambridge university press, 2003.

[267] R. T. Q. Chen, Y. Rubanova, J. Bettencourt et D. Duvenaud, « Neural ordinary differential equations », in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, sér. NIPS'18, Montréal, Canada : Curran Associates Inc., 2018, p. 6572-6583.

[268] C. Rackauckas, M. Innes, Y. Ma, J. Bettencourt, L. White et V. Dixit, *DiffEqFlux.jl - A Julia library for neural differential equations*, 2019. arXiv : 1902.02376 [cs.LG].

[269] J. Bontsema, J. Hemming, C. Stanghellini et al., « On-line estimation of the transpiration in greenhouse horticulture », English, in *Proceedings Agricontrol 2007. 2nd IFAC International Conference on Modeling and Design of Control Systems in Agriculture, Osijek, Croatia, 3 - 5 September, 2007*, IFAC, 2008, p. 29-34.

[270] W. J. Jo et J. H. Shin, « Effect of leaf-area management on tomato plant growth in greenhouses », *Horticulture, Environment, and Biotechnology*, **61**, *6*, p. 981-988, 2020. doi : 10.1007/s13580-020-00283-1.

[271] H. F. D. Zwart, « Analyzing energy-saving options in greenhouse cultivation using a simulation model », thèse de doct., Wageningen University et Research, 1996.

[272] E. M. Kleinbach, « Performance study of one-dimensional models for stratified thermal storage tank », thèse de doct., University of Wisconsin-Madison, 1990.

[273] E. Kleinbach, W. Beckman et S. Klein, « Performance study of one-dimensional models for stratified thermal storage tanks », *Solar Energy*, **50**, *2*, p. 155-166, 1993. doi : 10.1016/0038-092X(93)90087-5.

DOCTORAT BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Pilotage distribué de systèmes multi-énergies en réseau

Mot clés : Systèmes multi-énergies, commande prédictive, réseaux de neurones artificiels

Résumé : Un système multi-énergie se définit comme un ensemble de convertisseurs énergétiques permettant le couplage de différents vecteurs énergétiques (gaz, électricité, chaud, froid, etc.) pour répondre aux demandes d'unités de production, industrielles ou agricoles, voire d'immeubles abritant les activités humaines (logement, travail, loisirs) par exemple. La littérature actuelle foisonne de cas d'étude démontrant leur potentiel significatif pour des gains de flexibilité et d'efficacité énergétiques, moyennant la valorisation des synergies entre vecteurs et donc une commande optimale.

Ce travail contribue à la recherche relative à la problématique de la gestion optimisée des différents vecteurs énergétiques constitutifs des systèmes multi-énergies, au bénéfice de l'entité visée. Il embrasse à la fois la question de la modélisation et du pilotage. La formalisation du modèle s'appuie sur la notion d'*energy hub*. Son comportement dynamique est décrit par un modèle d'état neuronal. Afin de ne pas dépendre des seules données expérimentales, rarement disponibles en phase de conception, l'approche mise sur la complémentarité entre la modélisation multi-physique et la modélisation neuronale. La première procède par interconnexion de sous-systèmes souvent pré-existants dans les librairies ad hoc (cf. démarche générale de capitalisation de modèles multi-physiques). La seconde procède à partir de données, qui seront, dans le cadre de la méthodologie proposée ici, produite par la simulation du modèle multi-physique du système à l'étude. Notre réflexion ici a porté sur la question du choix d'une structure neuronale adaptée, et sur celle du choix des méta-paramètres d'apprentissage. Les modèles proposés ont ensuite été mis au service du pilotage, utilisés comme modèles internes de la commande dite prédictive ou EMPC (Economic Model Predictive Control). Cette dernière complète bien en effet la méthodologie globale, en ce qu'elle permet de gérer les compromis de pilotage, voire d'optimisation d'un coût économique. Le coût économique peut être relié notamment, au coût relatif des différents supports énergétiques).

Le cas d'application considéré in fine est celui du pilotage du système énergétique d'une serre agricole, comprenant comme vecteurs énergétiques de l'électrique, de la chaleur à basse et à haute température, du CO2 et comme source énergétique le gaz naturel. Les résultats obtenus montrent que la méthodologie développée est applicable et que l'utilisation de la commande avancée permet d' optimiser l'utilisation du gaz naturel par l'optimisation de l'utilisation des différents équipements du système énergétique de la serre.

Title: Distributed control of networked multi-energy systems

Keywords: Multi-energy systems, model predictive control, neural networks

Abstract: A multi-energy system is defined as a set of energy converters allowing the coupling of different energy carriers (gas, electricity, heat, cold, etc.) to meet the demands of production, industrial or agricultural units, or even buildings (housing, work, leisure) for instance. The current literature abounds with case studies demonstrating their significant potential for gains in flexibility and energy efficiency, through the enhancement of synergies between vectors and therefore optimal control.

This work contributes to research relating to the problem of the optimized management of the various energy carriers constituting multi-energy systems, for the benefit of the targeted entity. It embraces both the question of modeling and piloting. The formalization of the model is based on the notion of energy hub. Its dynamic behavior is described by a neural state model. In order not to depend on experimental data alone, rarely available in the design phase, the approach relies on the complementarity between multi-physical modeling and neuronal modeling. The first proceeds by interconnection of sub-systems often pre-existing in the ad hoc libraries (cf. general approach of capitalization of multi-physical models). The second proceeds from data, which will be, within the framework of the methodology proposed here, produced by the simulation of the multi-physics model of the system under study. Our reflection here focused on the question of the choice of an adapted neuronal structure, and on that of the choice of learning meta-parameters. The models proposed were then put to the service of piloting, used as internal models of the so-called predictive control or EMPC (Economic Model Predictive Control). The latter indeed complements the overall methodology, in that it makes it possible to manage control compromises, or even optimization of an economic cost. The economic cost can be linked in particular to the relative cost of the various energy supports).

The application case considered is that of controlling the energy system of an agricultural greenhouse, comprising electricity, low- and high-temperature heat, CO2 as energy carriers, and natural gas as an energy source. The results obtained show that the methodology developed is applicable and that the use of the advanced control makes it possible to optimize the use of natural gas by optimizing the use of the various equipment of the greenhouse energy system.