HAL
open science

Commande prédictive hybride et apprentissage pour la synthèse de contrôleurs logiques dans un bâtiment.

Duc Minh Khang Le

To cite this version:

Duc Minh Khang Le. Commande prédictive hybride et apprentissage pour la synthèse de contrôleurs logiques dans un bâtiment.. Autre. CentraleSupélec, 2016. Français. NNT : 2016CSUP0001 . tel-01646813

HAL Id: tel-01646813

https://theses.hal.science/tel-01646813v1

Submitted on 23 Nov 2017

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.

N° d'ordre : 2016-01-TH

CentraleSupélec

Ecole Doctorale MATISSE

« Mathématiques, Télécommunications, Informatique, Signal, Systèmes Electroniques »

Laboratoire IETR - UMR 6164 / équipe ASH

THÈSE DE DOCTORAT

DOMAINE : STIC
Spécialité : Automatique

Soutenue le 09 février 2016

par :

Minh Khang LE DUC

Commande prédictive hybride et apprentissage pour la synthèse de contrôleurs logiques dans un bâtiment

Composition du jury :

Directeur de thèse :	Hervé GUÉGUEN	Professeur IETR, CentraleSupélec
Co-encadrant	Romain BOURDAIS	Enseignant-Chercheur IETR, CentraleSupélec
Co-encadrant	Jacques PLÉVIN	Ingénieur-Expert DELTA-DORE
Présidente du jury :	Claire VALENTIN	Professeure LAGEP, Univ. Lyon
Rapporteurs :	Nicolas LANGLOIS	Professeur ESIGELEC
	Stéphane PLOIX	Professeur G-SCOP, Univ Grenoble-Alpes

Xin gui den gia dinh thân yêu ...

"Everybody is a genius. But if you judge a fish by its ability to climb a tree, it will live its whole life believing that it is stupid."

ALBERT EINSTEIN

Remerciements

Ce travail a été réalisé à CentraleSupélec, sur le Campus de Rennes, au sein de l'équipe Automatique des Systèmes Hybrides (ASH), en collaboration avec Delta Dore.

Je tiens à exprimer tout d'abord ma gratitude envers mes encadrants qui m'ont offert l'opportunité de travailler sur un sujet aussi intéressant que celui-ci et qui m'ont beaucoup soutenu, notamment vers la fin de la thèse. Je remercie ainsi :

— Mon directeur de thèse Monsieur Hervé Guéguen qui m'a permis de réaliser ma thèse dans d'excellentes conditions. Je le remercie également pour sa très grande disponibilité et sa gentillesse.

— Mon co-encadrant Monsieur Jacques Plévin dont sa très grande expérience est un véritable atout. J'ai particulièrement apprécié sa gentillesse et je suis très heureux de pouvoir continuer travailler avec lui à Delta Dore.

— Mon co-encadrant Monsieur Romain Bourdais pour son aide précieuse et sa patience au cours de ces 3 dernières années. Sa connaissance du monde antique, notamment les "Sept Merveilles" du monde m'a bénéficié pendant les heures de repos de midi.

J'adresse ensuite mes remerciements à Monsieur Nicolas Langlois et Monsieur Stéphane Ploix qui m'ont fait l'honneur d'être les rapporteurs de ma thèse. Je leur remercie pour leur attention qu'ils ont portée à la lecture de mon manuscrit. Notre échange était aussi très intéressant durant la soutenance, grâce à leur expertise et leur ouverture scientifique. Mes remerciements vont encore à Madame Claire Valentin pour avoir accepté d'être examinatrice de ce travail.

J'aimerais remercier l'équipe d'enseignants à CentraleSupélec : Monsieur Jean Buisson, Madame Marie-Anne Lefebvre, Monsieur Hervé Cormerais, Monsieur Pierre-Yves Richard, Monsieur Nabil Sadou avec qui j'ai pu travailler et avoir des échanges très enrichissants et intéressants.

Merci également à mes collègues thésards : Adrian, Antoine, Ilham, Maxime, Kate, Eunice, Safa, Soma, Jaehwa etc. pour l'ambiance sympathique durant ces dernières années.

J'adresse ensuite un grand merci à tous mes amis pour les bons moments passés ensemble, repas, ballades et conversations (Misa, XAnh, Dai Su, Khanh Hung, Bao Bao, Xuân Xuân, etc.). Leur amitié m'a été précieuse pour poursuivre cette thèse.

J'en termine par le plus important, mes parents Liêm, Mai, ma sœur Trang qui m'ont toujours permis de poursuivre mes études dans les meilleures conditions et ma femme Thao, qui m'a supporté et qui m'a beaucoup encouragé. Sans eux rien de cela ne serait possible, et je les en remercie infiniment.

Résumé

Une utilisation efficace et coordonnée des systèmes installés dans le bâtiment doit permettre d'améliorer le confort des occupants tout en consommant moins d'énergie. Ces objectifs à optimiser sont pourtant antagonistes. Le problème résultant peut être alors vu comme un problème d'optimisation multicritères. Par ailleurs, pour répondre aux enjeux industriels, il devra être résolu non seulement dans une optique d'implémentation simple et peu coûteuse, avec notamment un nombre réduit de capteurs, mais aussi dans un souci de portabilité pour que le contrôleur résultant puisse être implanté dans des bâtiments d'orientation différente et situés dans des lieux géographiques variés.

L'approche choisie est de type commande prédictive (MPC, Model Predictive Control) dont l'efficacité pour le contrôle du bâtiment a déjà été illustrée dans de nombreux travaux, elle requiert cependant des efforts de calcul trop important. Cette thèse propose une méthodologie pour la synthèse des contrôleurs, qui doivent apporter une performance satisfaisante en imitant les comportements du MPC, tout en répondant à des contraintes industriels. Elle est divisée deux grandes étapes :

1. La première étape consiste à développer un contrôleur MPC. De nombreux défis doivent être relevés tels que la modélisation, le réglage des paramètres et la résolution du problème d'optimisation.
2. La deuxième étape applique différents algorithmes d'apprentissage automatique (l'arbre de décision, AdaBoost et SVM) sur une base de données obtenue à partir de simulations utilisant le contrôleur prédictif développé. Les grands points levés sont la construction de la base de données, le choix de l'algorithme de l'apprentissage et le développement du contrôleur logique.

La méthodologie est appliquée dans un premier temps à un cas simple pour piloter un volet, puis validée dans un cas plus complexe : le contrôle coordonné du volet, de l'ouvrant et du système de ventilation.

Mots-clés : Hybrid MPC, Apprentissage automatique, Arbre de décision, SVM, Adaboost, Optimisation du confort, Pilotage du volet, Pilotage de la ventilation

Abstract

An efficient and coordinated control of systems in buildings should improve occupant comfort while consuming less energy. However, these objectives are antagonistic. It can then be formulated as a multi-criteria optimization problem. Moreover, it should be solved not only in a simple and cheap implementation perspective, but also for the sake of adaptability of the controller which can be installed in buildings with different orientations and different geographic locations.

The MPC (Model Predictive Control) approach is shown well suited for building control in the state of the art but it requires a big computing effort. This thesis presents a methodology to develop logical controllers for equipments in buildings. It helps to get a satisfactory performance by mimicking the MPC behaviors while dealing with industrial constraints. Two keys steps are required :

1. In the first step, an optimal controller is developed with hybrid MPC technique. There are challenges in modeling, parameters tuning and solving the optimization problem.
2. In the second step, different Machine Learning algorithms (Decision tree, AdaBoost, SVM) are tested on database which is obtained with the simulation with the MPC controller. The main points are the construction of the database, the choice of learning algorithm and the development of logic controller.

First, our methodology is tested on a simple case study to control a blind. Then, it is validated with a more complex case : development of a coordinated controller for a blind, natural ventilation and mechanical ventilation.

Keywords : Hybrid MPC, Machine learning, Decision tree, SVM, Adaboost, Comfort optimisation, Blind control, Ventilation control

Table des matières

I Introduction **1**

1 Contexte **3**
1.1 Contexte . . . 3
1.1.1 Confort thermique . . . 3
1.1.2 Qualité de l'air intérieur - QAI . . . 4
1.1.3 Confort visuel . . . 5
1.1.4 Confort acoustique . . . 5
1.1.5 Le confort, la consommation énergétique et le contrôle commande des équipements - le lien . . . 6
1.2 Objectif de la thèse . . . 7
1.3 Plan . . . 9
1.4 Publications . . . 10

2 Problématique **13**
2.1 Contrôleur logique . . . 14
2.2 Démarche . . . 14
2.3 HMPC . . . 15
2.3.1 Introduction au HMPC pour le batiment . . . 16
2.3.1.1 Objectif de fonctionnement . . . 16
2.3.1.2 Modèle . . . 17
2.3.1.3 Problème d'optimisation . . . 17
2.3.1.4 Méthode de résolution . . . 18
2.4 Machine Learning . . . 19
2.4.1 Petite introduction . . . 19
2.4.2 La base de données . . . 20
2.4.3 Décision . . . 20
2.4.4 Modèle . . . 21
2.4.4.1 Arbre de décision . . . 21
2.4.4.2 Boosting - AdaBoost . . . 23
2.4.4.3 SVM . . . 25
2.4.5 Over-Fitting . . . 27
2.4.6 Feature selection . . . 27
2.4.7 Weka . . . 28
2.5 Environnement de simulation-Introduction . . . 28
2.5.1 Présentation SIMBAD . . . 29
2.5.2 Cadre de travail . . . 29
2.6 Conclusion du chapite - Synthèse . . . 30

II Premier cas d'étude simple : Gestion des occultations **33**

3 Développement d'un contrôleur optimal avec la commande prédictive hybride HMPC **35**
3.1 Contexte et motivation 35
3.2 Développement de la commande prédictive hybride 36
3.2.1 Modèles 37
3.2.1.1 Modèle de température intérieure du bâtiment 37
3.2.1.2 Modèle de la luminosité 38
3.2.2 Traduction mathématique des critères et des contraintes 38
3.2.2.1 Critère de confort thermique 39
3.2.2.2 Critère de confort visuel 40
3.2.2.3 Critère de confort sonore et de la durée de vie des volets 40
3.2.3 Problème d'optimisation 40
3.2.4 Résolution du problème 41
3.3 Résultats de simulation 42
3.3.1 Conditions de simulation 43
3.3.2 Prévisions météorologiques parfaites 43
3.3.3 Prévisions météorologiques erronées 44
3.3.4 Comparaison avec un contrôleur classique 45
3.3.5 Économie sur la consommation énergétique 45
3.4 Conclusion 47

4 Développement d'un contrôleur simple grâce à un apprentissage automatique **49**
4.1 Motivation 49
4.2 Formation d'une base de données 50
4.3 Développement d'un contrôleur logique 50
4.3.1 Apprentissage avec Arbre de décision 50
4.3.2 Apprentissage avec AdaBoost 50
4.3.3 Apprentissage avec SVM 50
4.3.4 Comparaison entre différentes méthodes 53
4.3.5 Etude avec SVM 54
4.3.5.1 Performance du contrôleur avec différents nombres d'entrées . . . 55
4.3.5.2 Sensibilité des données 56
4.3.5.3 Adaptation des seuils en fonction du besoin de l'occupant 59
4.3.5.4 Simulation pendant une longue période 60
4.3.5.5 Portabilité de la solution 60
4.4 Conclusion 61

III Cas d'application : Gestion des occultations et de la ventilation **63**

5 Commande prédictive pour la gestion de la ventilation et des volets **65**
5.1 Introduction 65
5.1.1 Contexte 66
5.1.2 Objectif 67

5.2 Développement d'un contrôleur optimal MPC 67
5.2.1 Modèles 67
5.2.1.1 Modèle thermique correspond au fonctionnement à vitesse 1 de la ventilation mécanique 69
5.2.1.2 Modèle thermique correspond au fonctionnement à vitesse 2 de la ventilation mécanique 69
5.2.1.3 Commutation de modèles 69
5.2.1.4 Modèle de ventilation naturelle 71
5.2.1.5 Modèle du CO2 73
5.2.1.6 Modèle de luminosité 75
5.2.2 Traduction mathématique des critères et des contraintes 75
5.2.2.1 Critères pour la gestion des occultants 76
5.2.2.2 Critères pour la gestion de la ventilation 76
5.2.3 Problème d'optimisation 77
5.2.4 Résolution du problème 79
5.3 Résultats de simulation 80
5.3.1 Conditions de simulation 81
5.3.2 Période de mi-saison 81
5.3.2.1 Prévision parfaite 81
5.3.2.2 Prédictions imparfaites 83
5.3.2.3 Nombre d'occupants 84
5.3.3 Période d'été 85
5.4 Conclusion 86

6 L'apprentissage du comportement du contrôleur optimal 89
6.1 Introduction 89
6.2 Création d'une base de données 90
6.3 Développement d'un contrôleur logique 91
6.3.1 Sélection des attributs 91
6.3.1.1 Pour le système des occultations 92
6.3.1.2 Pour le système de ventilation 92
6.3.2 Apprentissage 93
6.3.2.1 Apprentissage pour les occultations 93
6.3.2.2 Apprentissage pour la ventilation 94
6.3.3 Résultats de simulation 95
6.3.3.1 Mi-saison 95
6.3.3.2 Eté 99
6.4 Conclusion 100

IV Conclusions et perspectives 103

Liste de figures 109

Liste des tables 113

Première partie

Introduction

CHAPITRE 1

Contexte

Sommaire

1.1 Contexte **3**
1.1.1 Confort thermique 3
1.1.2 Qualité de l'air intérieur - QAI 4
1.1.3 Confort visuel 5
1.1.4 Confort acoustique 5
1.1.5 Le confort, la consommation énergétique et le contrôle commande des équipements - le lien 6
1.2 Objectif de la thèse **7**
1.3 Plan **9**
1.4 Publications **10**

1.1 Contexte

Les usagers méritent leur place au cœur des enjeux de l'efficacité énergétique au sein du bâtiment !

En effet, nous passons de plus en plus de temps dans des espaces "fermés" tels que le logement, les écoles, les lieux de travail, etc. Une étude récente montre que la moyenne du temps passé à l'intérieure, dans des espaces clos est près de 80% chez les Français [1]. C'est pourquoi, un projet d'efficacité énergétique ne peut plus être pensé sans être lié à la notion de confort des occupants. L'importance de la notion de confort des usagers doit être reconnue.

Mais comment définissons-nous le confort ?

Le confort est défini comme un état de satisfaction, pour un environnement agréable dans lequel nous sommes affectés. Le confort dépend de trois facteurs principaux : physique de l'objectif environnemental ; physiologie du corps humain et psychologie humaine. De là, il faut souligner que le confort est une notion subjective, car elle varie d'une personne à une autre, selon l'état, les habitudes de chaque l'être humain. Assurer le confort correspond donc à satisfaire différents sens du corps humain : visuel, olfactif, auditif et tactile.

Ainsi, le confort comprend quatre éléments dont plus de détails sont présentés par la suite.

1.1.1 Confort thermique

C'est un état de satisfaction à propos de la température du corps humain. Le confort thermique est réalisé lorsqu'il existe un équilibre entre l'homme et l'ambiance, qui dépend de six facteurs liés à l'échange de chaleur entre l'homme et l'environnement.

FIGURE *1.1: Six facteurs affectent le confort thermique [http ://labo-energetic.eu]*

La figure 1.1 montre ces principaux facteurs parmi lesquels deux facteurs sont liés au comportement de l'occupant :

— L'habillement : la résistance thermique de vêtements (isolation des vêtements) qui représente le degré d'isolation entre la surface du corps et l'environnement.
— La physiologie thermique : la chaleur produite à l'intérieur du corps humain pour créer température d'équilibre, la chaleur réfléchie par physiologique (taux métabolique), en fonction de la posture, des activités humaines (assis, debout , marche, ...).

et quatre facteurs sont liés aux conditions environnementales :

— La température de l'air
— La température des parois
— La vitesse de l'air
— L'humidité

En particulier, il faut souligner l'importance de l'humidité relative dans la définition du confort thermique. Ainsi, une plage de confort hygrothermique peut être définie par le diagramme 1.2 suivant [2] :

Quatre principales zones peuvent être observées :

— Zone 1 : Zone à éviter vis-à-vis des problèmes de sécheresse.
— Zone 2 et 3 : Zones à éviter vis-à-vis des développements de bactéries et de micro-champignons.
— Zone 3 : Zone à éviter vis-à-vis des développements d'acariens.
— Zone 4 : Polygone de confort hygrothermique

Assurer un meilleur confort thermique est donc se situer le plus souvent possible dans la zone 4 du diagramme ci-dessous.

1.1.2 Qualité de l'air intérieur - QAI

Le but est d'amener de l'air frais à l'intérieur du bâtiment et d'éliminer des toxines, des polluants. En effet, l'air intérieur du bâtiment peuvent contenir des gaz très toxiques (e.g. CO, CO_2,

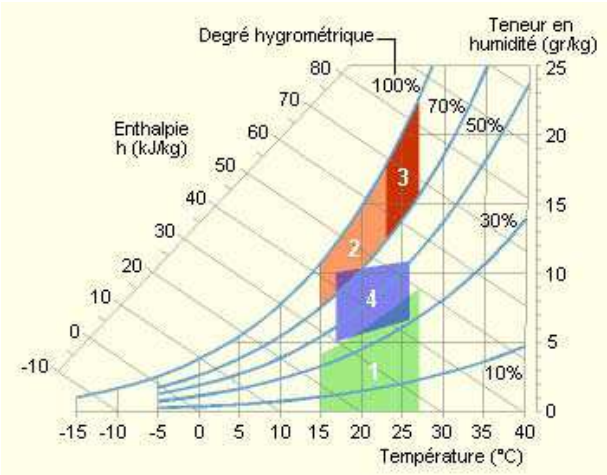

FIGURE *1.2: Différents plages de confort hygrothermique [http ://www.energieplus-lesite.be]*

le radon, la fumée, ...) ainsi que de la poussière, des moisissures, des bactéries, etc. Faute d'une ventilation adéquate, la qualité d'air sera sérieusement menacée, même nuisible pour la santé de l'occupant (causant de la dyspnée, de la fatigue, des maux de tête etc.).

1.1.3 Confort visuel

La lumière doit être répartie harmonieusement et suffisamment dans le bâtiment afin de créer un sentiment agréable à l'œil. De par une conception d'éclairage appropriée (puissance d'éclairage en conformité avec l'utilisation prévue de sites, distribution de lumière harmonique, ...), nous pouvons atteindre le confort visuel. De façon simplifiée, le confort visuel peut être caractérisé par une mesure en lux. Par exemple, la norme NF EN 12464 impose un éclairement minimal de 500 lux dans une salle de conférence, ou de 300 lux dans les salles de classe, etc. Dans une maison résidentielle, le confort visuel correspond en général à un minimum de 300 lux.

1.1.4 Confort acoustique

Il s'agit de créer un espace avec une bonne isolation acoustique pour que l'occupant puisse entendre clairement le son utile et ne pas être dérangé par les bruits provenant de l'extérieur du bâtiment tels que les bruits des véhicules, des bâtiments autour ou même des moteurs des équipements dans la maison (tels que chaufferie, ascenseurs, ventilation mécanique, etc.). En France, la réglementation actuellement en vigueur (arrêté du 30 juin 1999 relatif aux caractéristiques acoustiques des bâtiments d'habitation) fixe des exigences d'isolation acoustique entre logements. Par exemple, en matière d'isolement aux bruits extérieurs en façade, l'exigence minimale est 30 dB et peut en fait atteindre 45 dB en fonction du niveau sonore subi.

1.1.5 Le confort, la consommation énergétique et le contrôle commande des équipements - le lien

Afin de répondre aux besoins quotidiens des occupants ainsi que d'assurer les conforts que nous venons de détailler, différents systèmes ont été conçus pour être utilisés dans les buts suivants :

— Maintenir la température appropriée en hiver (chauffage) et en été (refroidissement) ;
— Renouveler l'air intérieur ;
— Répondre à la demande d'eau chaude ;
— Répondre aux besoins d'éclairage ;
— Répondre aux besoins des activités humaines (équipements électriques) ;
— etc.

Mais leur utilisation consomme de l'énergie. Pourtant, nous pourrions réduire de manière significative le niveau de consommation énergétique des bâtiments en étudiant leur construction (e.g. la forme, la taille, l'orientation des bâtiments, l'utilisation de l'isolation, etc.)

Ces derniers points ont été particulièrement traités dans la Règlementation Thermique 2012 (RT2012 [3]). En effet, la performance de l'isolation thermique et l'efficacité de l'étanchéité à l'air deviennent des points centraux pour atteindre l'objectif de basse consommation énergétique imposé par la RT2012. De plus, l'orientation des façades (en priorité au sud) et un taux minimal des surfaces vitrés (1/6 de la surface habitable) ont été aussi imposés afin de profiter au maximum des apports solaire gratuits pour chauffer en hiver.

Ces deux éléments permettent bien sûr de minimiser les besoins de chauffage dans les périodes d'hiver, mais génèrent des situations d'inconfort à la mi-saison comme en été à cause de l'inertie du bâtiment et de la difficulté d'évacuer de la chaleur déjà stockée à l'intérieur. De plus, l'étanchéité à l'air du bâtiment étant extrêmement renforcée, assurer un bon confort de la qualité de l'air intérieure devient difficile vu que les besoins de ventilation sont très souvent sous estimés.

C'est pourquoi, il est nécessaire de considérer le bâtiment avec ses systèmes thermiques et énergétiques comme un ensemble unique qu'il faut faire fonctionner de façon optimale. L'assistance de stratégie de contrôle « intelligent » assurant la régulation, l'optimisation des consommations devient dès lors indispensable. Pourtant, il ne faut pas oublier le comportement de l'occupant qui sera aussi un élément décisif. En effet, l'ensemble des systèmes présents dans le bâtiment sont là pour satisfaire les besoins de l'occupant, en termes de conforts thermique, acoustique, sanitaire, visuel, avec une bonne qualité de l'air et assurer sa sécurité. En parallèle, les systèmes de contrôle-commande doivent être fiables, faciles à utiliser et adaptables aux différents besoins de l'occupant.

Par conséquent, le challenge le plus important est de bien coordonner tous les systèmes installés dans l'habitat et d'optimiser différents objectifs qui sont pourtant antagonistes : réduire la facture énergétique et améliorer le confort. En effet, une utilisation efficace et coordonnée des systèmes installés dans le bâtiment (occultations, ventilation, émetteurs de chauffage, etc.) doit permettre de limiter les apports et gérer des transferts des zones fraîches vers les zones chaudes et ainsi de consommer moins d'énergie et d'améliorer le confort des occupants. Par ailleurs, pour répondre aux enjeux industriels, le problème devrait être résolu non seulement dans une optique d'implémentation simple et peu coûteuse, avec notamment un nombre réduit de capteurs, mais aussi dans un souci de portabilité pour que le contrôleur résultant puisse être implanté dans des

bâtiments d'orientation différente et situés dans des lieux géographiques variés.

1.2 Objectif de la thèse

L'objectif de notre travail se concentre sur le développement de stratégies de pilotage des équipements dans le bâtiment tout en assurant une bonne coordination de l'ensemble. C'est pourquoi, il est nécessaire, en premier lieu, d'avoir une structuration du système de contrôle-commande pour prendre en compte l'hétérogénéité des systèmes qui composent un bâtiment et leurs interactions. Une telle structure est présentée par la figure 1.3. Il est important de souligner que différents systèmes dans la maison (en rouge) sont connectés les uns avec les autres via un ou plusieurs critères (en vert) auxquels ils sont liés. Par exemple, la ventilation interagit avec les ouvrants par le critère "Qualité d'air". En effet, si avec un fonctionnement à grande vitesse de la ventilation mécanique ne suffit pas pour bien évacuer et faire baisser le taux CO_2 dans la pièce, il est nécessaire d'ouvrir l'ouvrant pour ventiler naturellement. L'information d'ouverture peut alors être transmise au système de ventilation qui fonctionnera à petite vitesse ou qui s'arrête complètement pour ne pas consommer d'électricité.

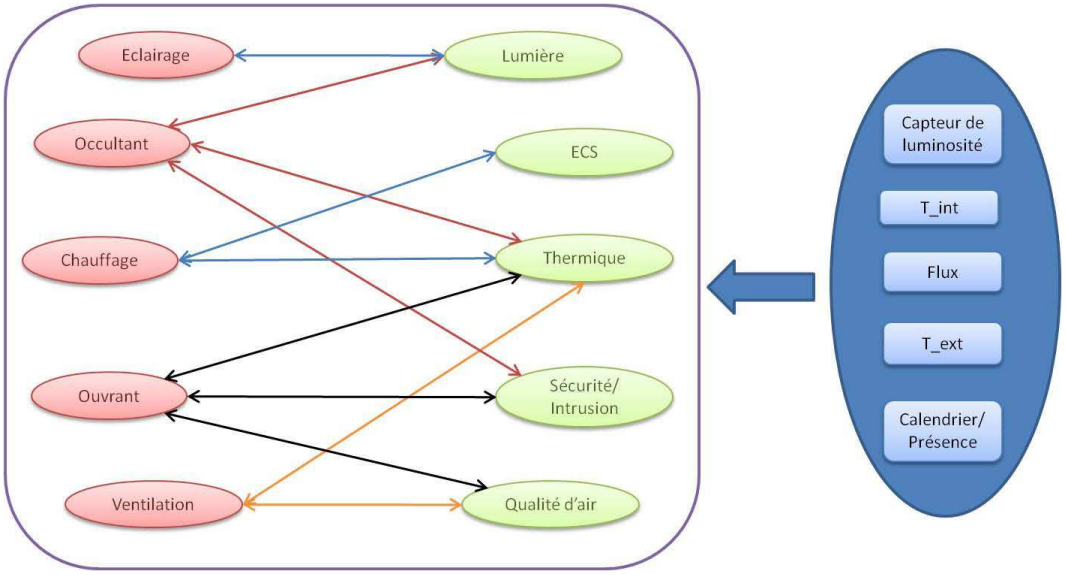

FIGURE *1.3: Proposition d'une structuration abstraite du fonctionnement de l'ensemble des systèmes dans l'habitat*

Le fait de raisonner comme décrit ci dessus simplifiera énormément l'ajout ou le retrait d'un système de l'architecture sans détériorer le bon fonctionnement d'autres systèmes restant. Il est à noter que l'information est fournie à l'ensemble des systèmes et critères par un réseau de capteurs (de température, de présence etc.) pour les aider à prendre leur décision.

La question se pose maintenant de savoir comment construire les algorithmes de contrôles et prendre de décision pour nos systèmes. A l'heure actuelle, les stratégies de pilotage construites avec des règles logiques de type "si ... alors ..." restent encore un des meilleurs choix pour les in-

génieurs du domaine de la GTB/GTC [4]. L'avantage de cette approche est la simplicité de sa mise en œuvre sur des automates programmables. Elle favorise la compréhension et facilite l'explication du fonctionnement des stratégies de commande. Pourtant, les bâtiments et les algorithmes de pilotage sont devenus de plus en plus complexes. En effet, chaque règle logique implique un certain nombre de paramètres qui doivent être ajustés, très souvent de façon intuitive et il devient compliqué d'assurer un bon fonctionnement des algorithmes de commande pour un système qui est construit avec un grand nombre de règles, par une bonne coordination de l'ensemble de différents systèmes.

Au cours des dernières années, la technique de commande prédictive MPC (Model Predictive Control) [5] attire particulièrement l'attention des chercheurs et des ingénieurs afin de contrôler les équipements dans la maison et elle propose ainsi un outil alternatif au contrôleur logique [6, 7, 8]. En effet, il s'agit d'une technique basée sur l'utilisation d'un modèle du bâtiment et des équipements ainsi que la prévision météorologique et de la présence des occupants sur un horizon de prédiction afin de déterminer une commande optimale. L'efficacité de la commande prédictive pour le contrôle du bâtiment a déjà été illustrée dans de nombreux travaux, elle requiert cependant des efforts de calcul trop importants pour être intégrée dans les produits industriels.

Il est clair que ni le MPC ni le contrôleur logique nous apporte une solution parfaite pour le pilotage de la maison mais le point fort de l'un permet de compléter et de remédier au point faible de l'autre. Notre but est de chercher à mettre en valeur la caractéristique optimale du MPC ainsi que la facilité du développement et du déploiement du contrôleur logique. En effet, l'objectif final de la thèse est de proposer une procédure d'extraction des règles logiques à partir des résultats du contrôleur optimal, et de pouvoir par la suite les implémenter dans un produit possédant une capacité de calcul limitée. Différentes étapes de notre procédure peuvent être représentée par le schéma suivant :

FIGURE *1.4: Schéma représentant différents étapes de notre méthodologie*

La première étape consiste à développer un contrôleur optimal de type MPC. Il est important de souligner le caractère hybride de ce dernier. En effet, nous ne nous substituons pas au propre contrôleur de chaque système, e.g. nous ne cherchons pas à réguler la puissance injectée d'un système de chauffage mais nous lui indiquons un mode de fonctionnement optimal qui pourrait être "hors gel", "arrêt", etc. Ensuite, une base de donnée est générée à partir des résultats de simulation et elle sert pour apprendre les comportements du contrôleur optimal avec des algorithmes du domaine d'apprentissage automatique. Enfin, un contrôleur simple est construit grâce à ces règles.

Notre méthodologie est tout d'abord étudiée pour un cas simple avec un seul système : le pilotage des volets pour assurer le confort thermique, sonore et visuel. Elle va ensuite être appliquée à un cas d'étude plus complexe avec le pilotage de l'ensemble des systèmes des occultations, de

la ventilation ainsi que de l'ouvrant dont l'architecture peut être présentée par la figure 1.5. Le but final est d'obtenir un contrôleur pour chaque système par l'apprentissage.

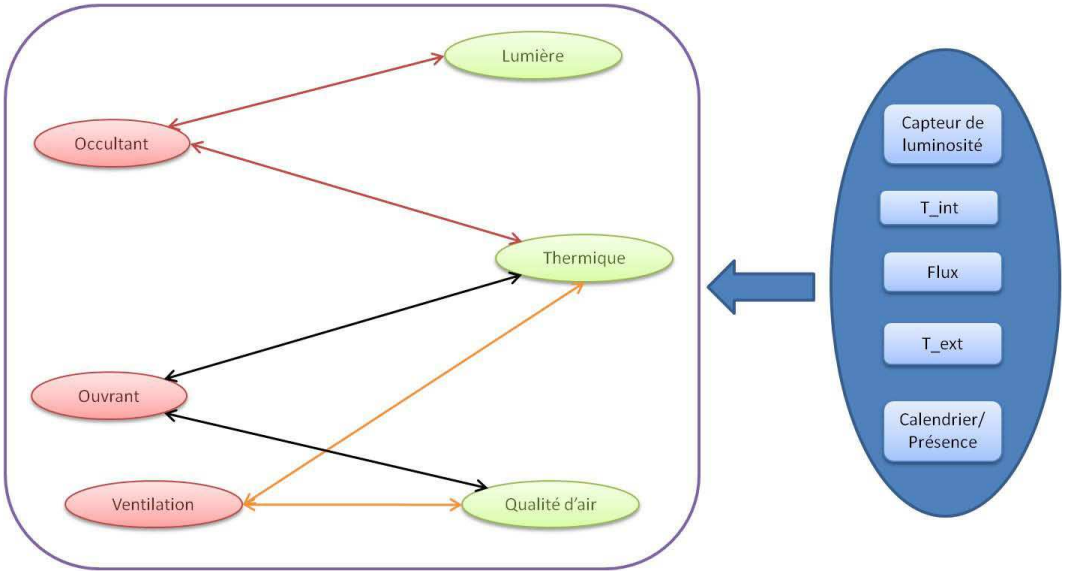

FIGURE *1.5: Schéma des interactions pour le pilotage de l'ensemble des occultations, de la ventilation et de l'ouvrant*

1.3 Plan

La suite de ce document est organisé comme suit :

Chapitre 2 Nous complèterons notre première partie "Introduction" en présentant l'état de l'art de différents éléments importants pour l'application de notre méthodologie. Nous commençons par la présentation du contrôleur logique. Ensuite, nous montrons comment et quels sont les éléments nécessaires afin de développer un contrôleur optimal MPC. Puis, nous présentons le domaine "Apprentissage Automatique", qui nous fournit des règles et des algorithmes pour comprendre le comportement du contrôleur optimal. Nous finissons par détailler notre cadre de travail avec la présentation de notre outil et notre maison de simulation.

Deuxième partie - Étude d'un cas simple pour piloter les occultations

Dans cette partie, nous présenterons l'application de notre méthodologie dans le cadre d'un cas simple pour contrôler un seul système, e.g. piloter les occultations. Nous divisons en deux chapitres qui correspondent à deux grandes étapes pour notre méthodologie.

Chapitre 3 Ce chapitre a pour but de développer un contrôleur prédictif pour la gestion des occultations afin d'assurer le confort thermique, visuel et sonore pour l'occupant. Nous présentons

ainsi des résultats de simulation avec ce contrôleur sous différents scénarios, i.e. le cas parfait et le cas imparfait faute d'information d'un capteur, pour évaluer sa pertinence.

Chapitre 4 L'objectif de ce chapitre est d'appliquer les différents algorithmes du domaine "Apprentissage automatique" aux données de simulations obtenues avec le MPC pour les occultations afin d'essayer de comprendre le comportement de ce dernier pour ensuite construire un contrôleur logique plus simple à implémenter. Ce dernier peut même apporter une méthode très simple pour s'adapter aux différents besoins de l'occupant, comme la consigne de température.

Troisième partie - Cas d'étude de l'ensemble du système de ventilation et des occultations

Cette partie présentera un cas d'étude plus complexe avec trois systèmes à piloter ensemble : le système de ventilation, l'ouvrant et les occultations. Elle est organisée de la même manière que la partie précédente avec deux chapitres.

Chapitre 5 Ce chapitre est réservé pour présenter le développement d'un contrôleur MPC pour contrôler de façon coordonnée le système de ventilation, de l'ouvrant et les occultations pour assurer le confort thermique, visuel, sonore et qualité d'air dans la maison. Les résultats obtenus fourniront des données pour le travail d'apprentissage.

Chapitre 6 Ce chapitre présente la deuxième étape de la procédure qui consiste à construire les contrôleurs logiques avec des techniques d'apprentissage automatique. Faut-il effectuer un apprentissage global ou un apprentissage de chaque système ? Les résultats de simulation effectués sur différentes saisons seront présentés pour évaluer la performance des méthodes d'apprentissage.

Conclusions et perspectives

Dans cette dernière partie, nous présentons un résumé de la thèse, nos conclusions ainsi que quelques perspectives pour les futurs travaux.

1.4 Publications

Articles

- LE, Khang, BOURDAIS, Romain, et GUÉGUEN, Hervé. From hybrid model predictive control to logical control for shading system : A support vector machine approach. Energy and Buildings, 2014, vol. 84, p. 352-359.
- LE, Khang, BOURDAIS, Romain, et GUÉGUEN, Hervé. Optimal control of shading system using hybrid model predictive control. In : Control Conference (ECC), 2014 European. IEEE, 2014. p. 134-139.

Brevet

— LE, Khang, BOURDAIS, Romain, GUÉGUEN, Hervé, PLÉVIN, Jacques. Method and device for controlling at least one device for concealing at least one opening of a room in a building. Brevet EP2682825. 06-07-2012.

— LE, Khang, BOURDAIS, Romain, GUÉGUEN, Hervé, PLÉVIN, Jacques. Method and device for controlling at least one device for concealing at least one opening of a room in a building. Brevet EP2682823. 06-07-2012.

CHAPITRE 2

Problématique

Sommaire

2.1 Contrôleur logique **14**
2.2 Démarche **14**
2.3 HMPC **15**
2.3.1 Introduction au HMPC pour le batiment 16
2.3.1.1 Objectif de fonctionnement 16
2.3.1.2 Modèle 17
2.3.1.3 Problème d'optimisation 17
2.3.1.4 Méthode de résolution 18
2.4 Machine Learning **19**
2.4.1 Petite introduction 19
2.4.2 La base de données 20
2.4.3 Décision 20
2.4.4 Modèle 21
2.4.4.1 Arbre de décision 21
2.4.4.2 Boosting - AdaBoost 23
2.4.4.3 SVM 25
2.4.5 Over-Fitting 27
2.4.6 Feature selection 27
2.4.7 Weka 28
2.5 Environnement de simulation-Introduction **28**
2.5.1 Présentation SIMBAD 29
2.5.2 Cadre de travail 29
2.6 Conclusion du chapite - Synthèse **30**

Nous allons apporter dans ce chapitre une description plus détaillée des éléments nécessaires pour notre méthodologie de construction d'un contrôleur logique à partir d'un apprentissage d'un contrôleur optimal.

Nous commençons par présenter le contrôleur logique suivi de différentes étapes de la démarche de notre méthodologie. La commande prédictive MPC avec ses éléments est présentée par la suite. Les outils nécessaires permettant d'apprendre ainsi que notre cadre de travail sont abordés à la fin du chapitre.

2.1 Contrôleur logique

Il s'agit d'une méthode de contrôle classique largement utilisée au sein de la communauté du contrôle des bâtiments. Elle est essentiellement basée sur un ensemble plus ou moins coordonné de règles logiques du type "si <condition> alors <action>". Les conditions sont généralement définies par un ingénieur expert, avec une bonne connaissance des phénomènes et des comportements du bâtiment. De nombreux brevets et des études scientifiques sur ce dispositif de contrôle avec l'application pour le pilotage des équipements pour l'objectif d'économie énergétique peuvent être trouvés dans [9, 10].

Un des avantages majeurs de cette approche est la simplicité dans sa mise en œuvre et elle est parfaitement adaptée pour être implémentée dans un produit commercial, par exemple dans des automates programmables ou être codée dans un micro processeur. De plus, le fonctionnement du contrôleur logique pourrait être représenté sous forme d'une arbre de décision dont chaque nœud est formé par une règle de décision. Ainsi, cela favorise la compréhension et facilite le travail des programmeurs.

En revanche, certains inconvénients de cette approche peuvent être cités comme :

— La définition des règles logiques ainsi que leur paramètres de réglages se fait généralement de façon intuitive. Il est difficile de justifier les valeurs des seuils intervenant dans chaque règle, notamment dans le monde du bâtiment où l'inertie de ce dernier doit être prise en compte et être anticipée.

— La concaténation d'un grand nombre de règles pourrait conduire à un grand arbre de décision et il devient donc difficile d'assurer un bon fonctionnement et une bonne cohérence de la commande.

— La structure du bâtiment devient de plus en plus complexe avec l'intégration de plusieurs systèmes avec des possibilités de production / stockage / revente et dans un contexte de prix de l'énergie variable. Dans ce cas de figure, l'utilisation d'un contrôleur logique atteint des limites.

2.2 Démarche

Face à des difficultés dans le choix des paramètres de réglages ainsi que le choix des entrées (i.e. des capteurs) pour la construction d'un contrôleur logique, il est préférable d'avoir une méthodologie pour mieux les choisir. Comme indiqué précédemment, ces choix et réglages sont souvent obtenus grâce à une bonne connaissance d'un expert dans le domaine. Par contre, l'expert est aussi humain, et l'humain ne peut pas tout faire.

La première étape de notre méthodologie consiste à remplacer l'expert par des contrôleurs optimaux parmi lesquels la commande prédictive MPC (Model Predictive Control) a beaucoup attiré l'attention des chercheurs ces dernières années. En effet, il s'agit d'une méthode de commande avancée qui nécessite un modèle du bâtiment et des prévisions (météorologiques ou de la présence de l'occupant) pour nous fournir la meilleure sortie du contrôleur afin d'assurer une sortie voulue du système. Pourtant, le défaut majeur de la commande MPC est qu'elle est particulièrement coûteuse en temps de calcul et il devient impossible de l'implémenter dans un micro contrôleur pour pouvoir commercialiser ou de le faire fonctionner en temps réel.

La deuxième étape consiste à apprendre les comportements de ce dernier et puis de construire un contrôleur logique à partir de ces connaissances. Le domaine Data Mining est parfaitement adapté pour cette démarche en proposant plusieurs outils avec une base mathématique solide.

Il est important de remarquer que les équipements dans la maison fonctionnent souvent suivant différents modes. Par exemple, la sortie du système de chauffage peut être sous forme de quatre modes de fonctionnement comme : Confort, Nuit, Hors Gel, Arrêt. De même, le système de ventilation mécanique peut fonctionner en mode 1 (vitesse nominale) ou en mode 2 (vitesse augmentée). C'est pourquoi, il faut faire face à la commande prédictive des systèmes de type hybrides. Le but est donc de trouver les meilleures sorties des systèmes mais nous ne nous intéressons pas ici au pilotage même des équipements.

La commande prédictive hybride ainsi que les techniques du domaine Data Mining vont être présentées plus en détail dans la partie suivante.

2.3 HMPC

Dans cette section, l'introduction à la commande prédictive va être présentée. Seuls les informations essentielles et les points clés dans la formulation du problème du MPC seront fournis par la suite.

Depuis sa première application dans les années 1970, l'utilisation de la commande prédictive s'est répandue industriellement à travers plusieurs champs d'activité (procédés chimiques, l'industrie alimentaire, l'automobile, etc. [11, 12]). Il s'agit d'une méthode de contrôle avancée en s'appuyant sur l'utilisation d'un modèle du processus qui synthétise la dynamique du procédé à commander.

La stratégie de commande prédictive MPC peut être décomposée en trois étapes :

1. La sortie du système à contrôler est prédite sur un horizon temporel futur grâce à un modèle approprié ;
2. Une séquence de contrôle est calculée en optimisant une fonction objective basée sur le comportement futur du système, est alors uniquement le premier élément de la séquence de contrôle appliqué ;
3. Le calcul est ensuite répété à l'instant suivant.

De nombreuses variantes de cette technique ont émergé dans les dernières décennies, on peut citer par exemple GPC (Generalized predictive control) [13], MPC Robuste [14], NMPC (Non Linear MPC) [15], etc. A cause du caractère hybride de nos systèmes à commander dans le bâtiment, nous visons de mettre en œuvre la commande prédictive hybride (HMPC). En effet, en considérant le problème de régulation de température intérieure dans la maison avec le système de chauffage, les états de notre modèle (e.g. la température) sont continus tandis que la sortie de contrôle est discrète (e.g. différents modes de fonctionnement du chauffage).

La section suivante va détailler un peu plus sur le HMPC pour l'application du pilotage des bâtiments, en particulier dans nos cas d'application. Plusieurs résultats dans ce domaine de recherche peuvent être trouvés dans [16, 17, 18]

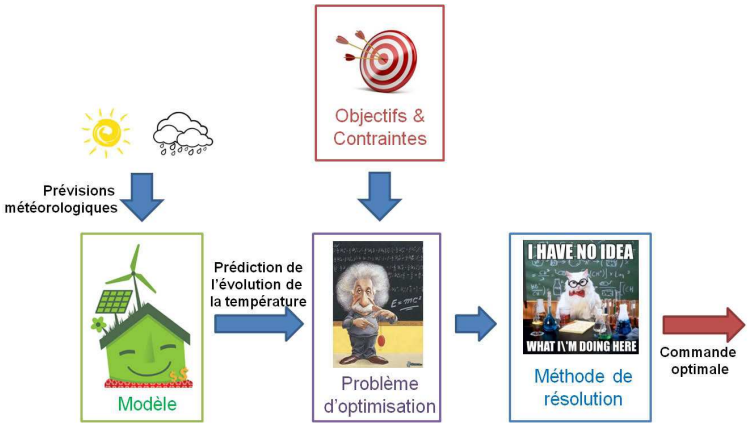

FIGURE *2.1: Différents éléments importants pour la formulation d'un problème de contrôle prédictif. Nous y trouvons les objectifs et les contraintes, les modèles de prédiction ainsi qu'un problème d'optimisation suivi d'une méthode de résolution*

2.3.1 Introduction au HMPC pour le batiment

Les ingrédients nécessaires pour une application du HMPC en général ou plus particulièrement dans le bâtiment sont présentés par la figure 2.1 où nous pouvons citer des points suivants :

— Un objectif de fonctionnement
— Un modèle du procédé à commander
— Un problème d'optimisation
— Un solveur, une méthode de résolution du problème d'optimisation

Chaque élément sera présenté dans la partie suivante.

2.3.1.1 Objectif de fonctionnement

A partir du cahier des charges, un objectif de fonctionnement peut se traduire par différents objectifs posés par les ingénieurs de design du contrôleur :

— Assurer différents conforts des occupants (cf. Section Confort). Par exemple, la température intérieure ne doit pas dépasser $26\,^{\circ}\mathrm{C}$ en été ou la luminosité dans la maison doit rester supérieure à 300 lux si possible, etc.
— Optimiser un critère de performance lié à des critères économiques, e.g. minimiser la consommation d'énergie dans le bâtiment, etc.

ainsi que des contraintes de fonctionnement que doivent respecter les équipements :

— Des contraintes liées aux équipements elles mêmes. Il faut limiter le nombre de mouvement du système des occultations par exemple, afin de conserver le cycle de vie du moteur.
— Des contraintes liées aux occupants. Il s'agit par exemple d'un fonctionnement uniquement en absence des utilisateurs afin de ne pas perturber leur vie.

2.3.1.2 Modèle

Pour mettre en œuvre la commande prédictive, il est nécessaire d'avoir un modèle qui permet de prédire l'évolution du comportement du procédé. Dans notre cas d'étude, un modèle du bâtiment qui nous fournira l'évolution de la température dans les heures à venir par exemple. Plusieurs types de modèle existent dans la littérature, on peut citer par exemple :

— Modèle de type boîte blanche : dans un modèle boîte blanche, les lois et les propriétés physiques sont connues. A partir de cette information, les paramètres du modèle sont ainsi obtenus. Le comportement du modèle est décrit par une connaissance complète du système, tels que les formules mathématiques, structures physiques et les valeurs de paramètres idéales des composants de modèle. Il faut noter que cette approche n'est pas souvent réaliste car il n'y a pas de connaissance idéale complète d'un système.

— Modèle de type boîte noire : dans les cas où il n'y a aucune information préalable sur la structure du modèle, ou sur l'interaction entre la structure et l'entrée/sortie du système, le modèle de type boîte noire est utilisé. Il est basé sur une structure de modèle général dans le domaine temporel ou dans le domaine fréquentiel.

— Modèle de type boîte grise : c'est la modélisation intermédiaire entre la modélisation de type boîte blanche et celle de type boîte noire. Même si la modélisation peut s'appuyer sur des lois physiques, les paramètres du modèle sont pourtant inconnus ou il y a des phénomènes qui ne sont pas faciles à décrire. Les éléments inconnus du modèle peuvent être néanmoins estimés à partir des mesures.

Pour nos applications, nous utiliserons des modèles boîte noires dont les détails se trouveront dans les parties 3.2.1 et 5.2.1.

2.3.1.3 Problème d'optimisation

Il s'agit de la formulation mathématique des objectifs à satisfaire sous les contraintes de fonctionnement du procédé. En effet, il revient à optimiser, sur un horizon de prédiction Np une fonction "critère" J qui est aussi appelée fonction coût qui dépend de :

— $x(k)$: l'état du système à l'instant k (e.g. la température intérieure, l'état du système des occultation s etc.)

— $D(k : k+Np-1)$: les perturbations (e.g. le flux solaire, la température extérieure etc.)

— $u_s(k : k+Np-1)$: la variable d'optimisation discrète (e.g. la position du volet ou le mode de fonctionnement du système de ventilation etc.)

Le problème se traduit par l'optimisation de la somme des critères dont l'exemple peut être :

— $J_{temp}[\hat{x}(k+1 : k+Np]$: un critère de température intérieure dans la maison, qui compare par exemple les prédictions de température avec un seuil de température

— $J_{lum}[D(k : k+N-1, u_s(k : k+Np-1]$: le critère du niveau luminosité dans la pièce

— etc.

et aussi des contraintes qui sont relâchées et qui peuvent être exprimées sous forme de critères, par exemple :

— $J_{mov}[u_s(k+1 : k+Np]$: le contrainte sur le nombre de changement de position du volet

— etc.

Il est important de noter que nous n'étudions pas le problème de faisabilité car il est parfois

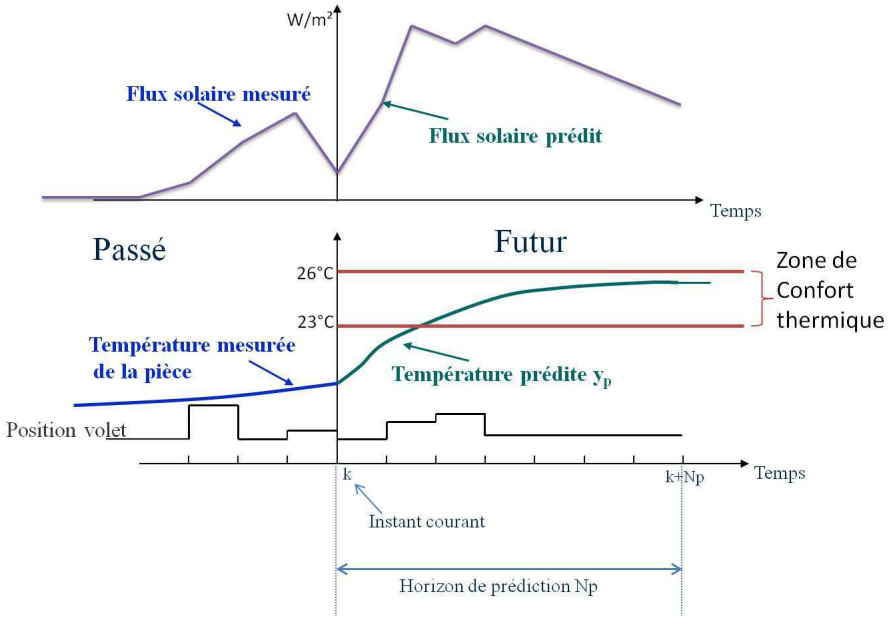

FIGURE *2.2: Principe de fonctionnement de la commande prédictive MPC*

impossible de satisfaire les critères de confort, i.e. la température continue à monter et dépasse le seuil de confort thermique malgré une fermeture complète des occultations. Dans ces conditions extrêmes, nous acceptons le non respect des contraintes afin de ne pas rendre le problème d'optimisation impossible à résoudre.

2.3.1.4 Méthode de résolution

Après avoir écrit le problème sous forme d'une fonction de coût, il faut maintenant le résoudre à chaque instant afin d'obtenir la meilleure séquence de commande sur l'horizon de prédiction Np.

Plusieurs outils performants sont proposés dans la littérature ainsi que dans l'industrie [19, 20, 21, 22] pour résoudre le problème d'optimisation dont les sorties du contrôleur sont discrètes, nous avons adopté la méthode Branch and Bound (B & B) [23], qui permet de trouver une séquence optimale des actions de contrôle en minimisant une fonction de coût.

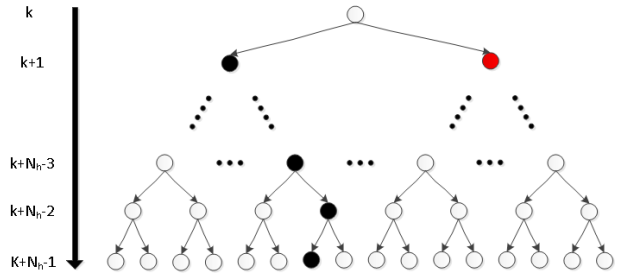

FIGURE *2.3: Explication du principe de fonctionnement de la méthode Branch & Bound*

L'idée principale est de développer toutes les séquences de contrôle possibles sous la forme de l'arbre (Branch) où N_p est le profondeur de l'arbre ainsi que l'horizon de prédiction. Chaque nœud de l'arbre correspond ainsi à la décision à chaque instant sur la durée de prédiction (cf. Figure 2.3). C'est pourquoi, le coût va accroître avec le profondeur de recherche. Considérons n le nombre de sorties du contrôleur, le nombre total des possibilités est n^{N_p}. Plus l'horizon de commande est grand, plus la recherche de la solution optimale devient difficile. Afin de réduire l'effort de calcul, la technique de découpage de l'arbre est mise en œuvre. L'idée est d'arrêter d'explorer le reste d'une branche lorsque son coût partiel dépasse une limite supérieure qui a été déjà établie dans une autre branche. La figure 2.3 montre la structure de l'arbre de l'algorithme B & B ainsi que l'illustration de la méthode de découpage.

Par exemple, l'optimiseur ne cherche pas à exploiter la suite de branche issue du point rouge si son coût est déjà supérieur au coût de la séquence des points noirs qui est la meilleure séquence de solution déjà trouvée, dans ce cas il est inutile de continuer la recherche dans la sous-arborescence enracinée en ce nœud, puisque les bornes inférieures obtenues sont strictement croissantes suivant la profondeur de l'arbre du "Branch and Bound".

Des lectures plus profondes sur cette méthode B & B pour l'application dans le domaine des systèmes hybrides se trouvent dans [24, 25, 26].

2.4 Machine Learning

La prochaine et dernière étape dans notre méthodologie est l'apprentissage des comportements du contrôleur optimal. C'est donc le but de cette section. Nous allons commencer par une introduction du domaine Machine Learning qui nous fournit des outils puissants pour l'analyse et l'interprétation de ces données et ainsi pour l'extraction de connaissances utiles qui peuvent nous aider à prendre une décision. Les outils ainsi que leur base mathématique seront présentés dans les dernières parties de ce chapitre.

2.4.1 Petite introduction

> "L'apprentissage automatique, c'est la capacité d'un ordinateur à apprendre sans avoir été explicitement programmé."
>
> (Arthur Samuel)

En effet, l'apprentissage automatique (ou Machine Learning en anglais [27]) est un domaine où le but est de créer de la connaissance de manière automatique à partir d'une base de données. Des décisions peuvent ensuite être prises à partir de l'exploitation de ces connaissances. De nombreuses applications utilisant des algorithmes d'apprentissage automatique existent, nous pouvons citer par exemple les systèmes qui analysent les données historiques de vente afin de prédire le comportement des clients, ou l'optimisation du comportement du robot afin qu'une tâche soit complétée en utilisant un minimum de ressources, ou encore l'extraction des connaissances à partir de données bioinformatiques, etc. [28]

Le principe du Machine Learning peut être expliqué par le schéma présenté par la figure 2.4 :

On distingue 3 étapes principales :

— Construire une base de données qui va ensuite être utilisée pour apprendre des modèles,
— Déterminer un modèle de décisions, à l'aide des algorithmes,

FIGURE *2.4: Principaux éléments d'un problème d'apprentissage automatique*

— Utiliser le modèle sur de nouvelles données pour prendre des décisions

Toujours dans le but de faire un lien avec notre cas d'étude, plus de détails sur ces points ainsi qu'un exemple et des explications plus concrètes seront présentés dans la partie suivante.

2.4.2 La base de données

La base de données est l'ensemble des échantillons qui sont souvent écrits sous forme de vecteur.

$$x = (x_1, x_2, \ldots, x_n)$$

où n est le nombre total des coordonnées qui est aussi appelé "attributs" ou aussi "features" dans le domaine Machine Learning.

Dans notre méthodologie, une simulation avec la commande optimale est effectuée afin de générer une base de données pour le problème d'apprentissage. Les états du système ainsi que les conditions externes peuvent être sauvegardés. On peut citer par exemple le flux solaire, la température extérieure, la température intérieure, etc.

Deux types d'attributs sont à distinguer :

— Attribut étiqueté : il s'agit d'un problème d'apprentissage supervisé dont les données sont liées à un label d'une décision. Il est clair que nos travaux se situent dans ce cadre car nous connaissons parfaitement à quel état de la sortie du commande correspondent nos données de sauvegarde.

— Attribut non-étiqueté : le problème d'apprentissage est ici non-supervisé pour des données qui ne sont pas accompagnées d'un étiquette. Ce type d'apprentissage se trouve souvent dans la reconnaissance de visage par exemple.

Il faut souligner qu'il est important de bien choisir les données pertinentes pour l'apprentissage car les données redondantes diminueront considérablement la qualité du modèle déduit. L'extraction de caractéristiques ("Feature Selection" en anglais) est un outil très efficace pour nous fournir un indice sur l'importance de ces attributs. Il est souvent lié à des algorithmes proposés par le Machine Learning. C'est pourquoi, nous reviendrons sur le détail de ce point un peu plus tard (cf. 2.4.6).

2.4.3 Décision

La décision est liée au type de sortie que l'on veut prédire. Il existe deux catégories de décision :

— Sortie discrète (catégorisé) : le nombre de valeurs est fini. Nous parlons alors de "classe". Puisque la sortie de la commande MPC est un mode de fonctionnement ou un état du système à contrôler, nous nous intéressons principalement à ce type de décision dans nos cas d'étude.

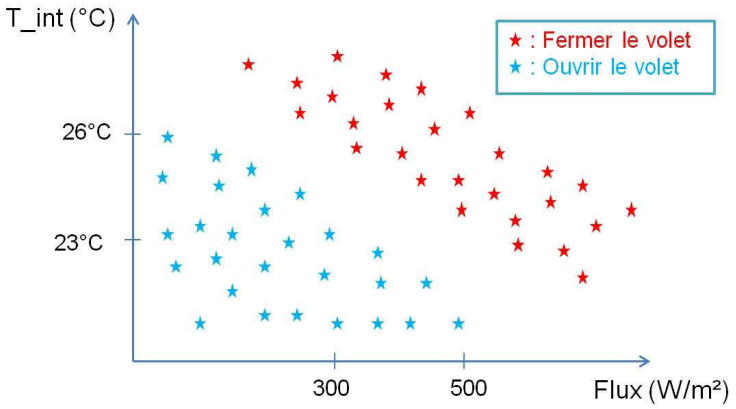

FIGURE *2.5: Exemple de présentation en deux dimensions des données enregistrées. Les décisions sont "fermer le volet" et "ouvrir le volet". Les attributs sont la température intérieure et le flux solaire*

— Sortie continue : contrairement au point précédent, le nombre de valeurs est ici infini.

Un exemple d'une sauvegarde des données suite à une simulation effectuée avec la commande prédictive pour le pilotage d'un volet est donné par la figure 2.5. La base de données est présentée dans une espace bidimensionnelle avec deux attributs pris en compte, i.e. les mesures de flux et les mesures de température extérieure. Chaque point est lié à une décision, e.g. soit le volet est fermé complètement, soit le volet est ouvert complètement.

L'ensemble de k points d'apprentissage qui sont attachés à deux attributs "Flux solaire" et "Température extérieure" ainsi qu'à deux décisions "Fermer" et "Ouvrir" peut être présenté par :

$$(x_{(i,Flux)}, x_{(i,Text)}, y_i)$$

avec $y_i \in Y$, $Y =$ {Fermé, Ouvert} pour i=1,. . .,k

2.4.4 Modèle

Il reste maintenant à trouver une fonction qui est capable de prédire la décision y d'une nouvelle observation x. Il existe plusieurs approches pour apprendre une telle fonction. Nous présenterons par la suite trois algorithmes les plus connus dans la littérature, suivis des exemples simples de leur application dans un problème d'apprentissage automatique :

— Arbre de décision
— Adaboost
— Machine à vecteurs de support (SVM)

Des détails de ces approches sont présentés dans la partie suivante :

2.4.4.1 Arbre de décision

La méthode de construction d'un arbre de décision est l'une des méthodes classiques la plus utilisée car son avantage est d'être aisément interprétable par un humain et d'être très rapidement

applicable par une machine. Elle est basée sur un découpage par des hyperplans parallèles aux axes de l'espace constitué par les attributs. Plusieurs algorithmes permettent de construire un arbre de décision parmi lesquels nous pouvons citer CART [29] ou C4.5 [30]

Principe de fonctionnement L'idée principale de la méthode est de diviser récursivement et le plus efficacement possible les exemples de l'ensemble d'apprentissage par des tests définis à l'aide des attributs jusqu'à ce que l'on obtienne des sous-ensembles d'exemples ne contenant que (ou presque) des exemples appartenant tous à une même classe.

Les étapes pour obtenir une arbre de décision sont :

— Décider si un nœud est terminal. Si tous les éléments associés à ce nœud sont dans une même classe ou si aucun test n'a pu être sélectionné ou si il y a moins d'un certain nombre d'erreurs alors un nœud est terminal.
— Sélectionner un test à associer à un nœud si ce dernier n'est pas terminal.
— Affecter une classe à une feuille. On attribue la classe majoritaire. Si il n'y a aucun exemple on attribue la classe majoritaire du nœud père.

Exemple La figure 2.6 montre le principe du découpage d'algorithme d'arbre de décision, appliqué à notre exemple simple du pilotage des occultations.

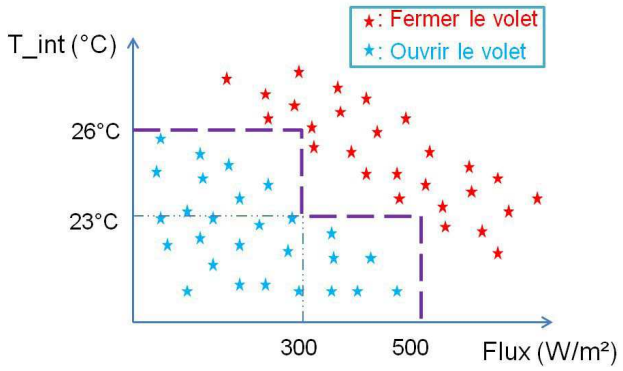

FIGURE *2.6: Principe du découpage d'algorithme d'arbre de décision avec notre exemple pour le pilotage des volets*

L'arbre de décision correspondant est décrit ci-dessous :

— La variable utilisée au premier sommet (la racine de l'arbre) est "la température intérieure". Avec un seuil établi à $26\,^{\circ}\mathrm{C}$, elle permet de décider qu'au dessus de ce seuil, le volet est fermé.
— La variable "le flux solaire" est prise ensuite avec un seuil qui est à 500 W/m^2. Ils permettent de décider qu'au delà de 500 W/m^2, le volet est fermé.
— La branche est développée avec la variable "température intérieure" et la décision d'ouvrir le volet si la température est inférieure à $23\,^{\circ}\mathrm{C}$.
— Enfin, il est décidé de fermer le volet si le flux dépasse 300 W/m^2. Dans le cas contraire, le volet est ouvert. Le développement de l'arbre s'arrête car elle a réussi à identifier tous les points de la base de données.

Un arbre peut être construit suite à des résultats du découpage comme montre la figure 2.7. Il est important de souligner que la solution apportée par la méthode de l'arbre de décision n'est pas unique.

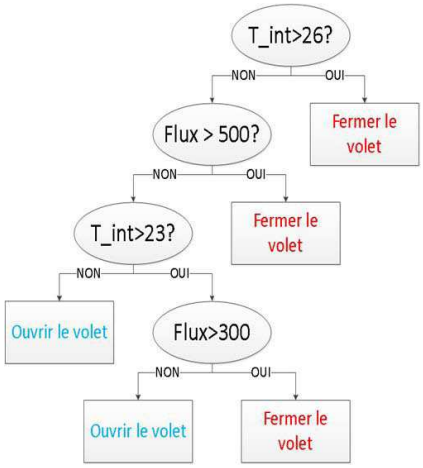

FIGURE *2.7: L'arbre de décision construit à partir des résultats du découpage précédent, avec notre exemple pour le pilotage des volets*

2.4.4.2 Boosting - AdaBoost

Le Boosting [31] est une méthode avancée se basant sur les arbres de décisions. Il s'agit d'un algorithme qui consiste à créer une règle de prédiction précise en combinant de nombreuses règles relativement faibles et inexactes. Ainsi, au lieu de former un classificateur avec des règles complexes, nous construisons des classificateurs extrêmement simples (avec une ou quelques règles de décision) et la prédiction est obtenue en pondérant les réponses de tous ces classificateurs.

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$$

avec :

— $h_t(x)$ est un classificateur simple (weak learner)

— α_t pondération du classificateur simple

— $H(x) = sign(f(x))$ est le classificateur final

Un des algorithmes les plus utilisés en Boosting s'appelle AdaBoost, qui signifie adaptative boosting et qui a été introduit par Freund et Schapire [32].

Le principe de base de AdaBoost est d'obtenir une séquence de classificateurs simples (weak learner, c'est-à-dire les modèles très simples qui sont par exemple un seuil ou un petit arbre de décision) en modifiant de façon répétitive les données à classifier. Les prédictions de chacun des weak learners sont ensuite combinées par une somme de vote pour produire la prédiction finale. Les modifications de données à chaque itération sont appelées "boosting" et elles consistent à

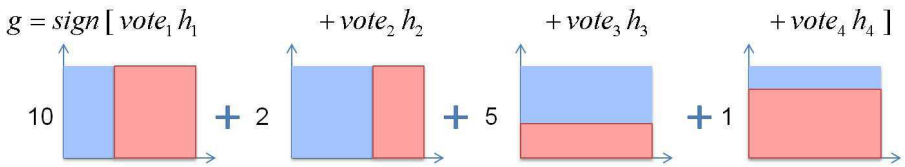

FIGURE *2.8: Exemple de l'application de la méthode d'AdaBoost. La décision finale est la somme du produit de toutes les classificateurs simples et leur pondération*

appliquer des poids α_1, α_2, ..., α_N à chacun des échantillons d'apprentissage. Initialement, ces poids sont tous mis à $\alpha_i = 1/N$, de sorte que la première étape entraîne simplement un "weak learner" sur les données d'origine. Pour chaque itération successive, les pondérations d'échantillon sont individuellement modifiées et l'algorithme d'apprentissage est à nouveau appliqué aux données re-pondérées. Les données qui ont été mal prédites vont voir leur poids associé au modèle augmenter, tandis que le poids est réduit pour celles qui ont été prédites correctement. Par conséquent, l'influence des exemples qui sont difficiles à prédire augmentera au fur et à mesure. Tous les weak learners sont ainsi forcés de se concentrer sur les exemples qu'ils n'ont pas réussi à classer correctement précédemment.

Description de l'algorithme Considérer un ensemble d'apprentissage $(x_1, y_1), \ldots, (x_k, y_k)$ avec k le nombre de données x, $x_i \in \Re^n$. Chaque exemple x_i appartient à une classe $y_i \in \{-1, +1\}$.

A l'initialisation, supposons $D_1(i) = 1/k, i = 1, \ldots, k$.

Pour $t = 1, \ldots, T$, avec T le nombre d'itérations prédéfini :

— Trouver $h_t = arg \min_{h_t \in H} \varepsilon_t = \sum_{i=1}^{k} D_t(i)[y_i \neq h_t(x_i)]$, $H = h(x)$

— Si $\varepsilon_t \geq 1/2$ alors arrêter

— Sinon, mettre $\alpha_t = \frac{1}{2} \log \frac{1 + r_t}{1 - r_t}$

avec $r_t = \sum_{i=1}^{k} D_t(i) h_t(x_i) y_i$

— Mettre à jour

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

avec Z_t est constante de normalisation, $Z_t = 2\sqrt{\varepsilon_t(1 - \varepsilon_t)}$

Le classificateur final s'écrit ainsi :

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

Exemple Les résultats donnés par l'algorithme AdaBoost pour notre cas d'exemple de la gestion des volets sont représentés avec la figure 2.8. Nous y trouvons un classifieur g qui se compose de quatre classifieurs simples (i.e. h_1, h_2, h_3, h_4) avec différentes pondérations (i.e. $vote_1$, $vote_2$, $vote_3$, $vote_4$).

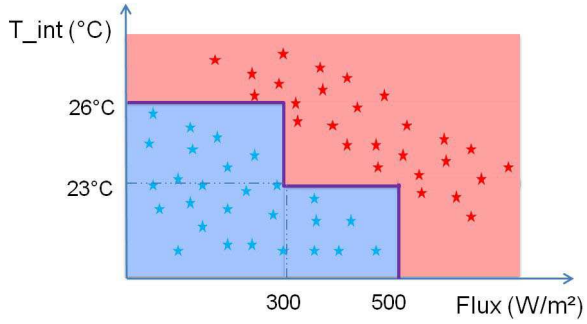

FIGURE *2.9: Représentation graphique du classificateur final obtenu avec la méthode d'Adaboost*

2.4.4.3 SVM

Cette section propose d'aborder une nouvelle famille d'algorithmes : les SVM ou (Support Vector Machines, traduit par Séparateurs à Vaste Marge ou machine à vecteur support) dont le principe s'appuie sur la notion de distance entre les données. En effet, il consiste à trouver la courbe qui « sépare au mieux » les données en maximisant la marge entre leurs classes.

Le SVM a été introduit par V. Vapnik [33] en se basant sur la théorie de la minimisation du risque structurel. La figure 2.10 montre l'exemple de l'utilisation du SVM pour séparer nos décisions de fermer ou de laisser ouvert notre volet. Nous y trouvons des vecteurs de supports qui sont entourés par des cercles et qui sont les plus proches de l'hyperplan séparant deux décisions.

De plus, l'hyperplan peut être non linéaire en utilisant des "feature functions" (ou méthodes à noyaux en français). Cette méthode est souvent utilisée dans le cas où les données sont non-séparables. Le principe est d'appliquer une transformation judicieuse sur les données afin d'augmenter leur dimension pour les rendre séparable linéairement. Il suffit ensuite d'appliquer la transformation inverse à l'hyperplan séparateur. Plus de lectures sur cette méthode se trouvent dans [34, 35, 36, 37].

Les bases mathématiques du SVM sont brièvement décrites dans la partie suivante pour le cas séparable et non séparable linéairement.

Description mathématique Considérer un ensemble d'apprentissage $(x_1, y_1), \ldots, (x_k, y_k)$ avec k le nombre de données x, $x_i \in \Re^n$. Chaque exemple x_i appartient à une classe $y_i \in \{-1, +1\}$.

Dans le cas linéairement séparable, il existe un vecteur w et un scalaire b qui satisfont les inégalités suivantes :

$$\begin{aligned} x_i w + b &\geq +1 \quad \text{pour } y_i = +1 \\ x_i w + b &\leq -1 \quad \text{pour } y_i = -1 \end{aligned}$$

qui peuvent être écrites sous une forme compacte :

$$y_i(x_i w + b) \geq +1 \quad i = 1, \ldots, k$$

La marge peut être exprimée comme :

$$d = \frac{2}{\|w\|}$$

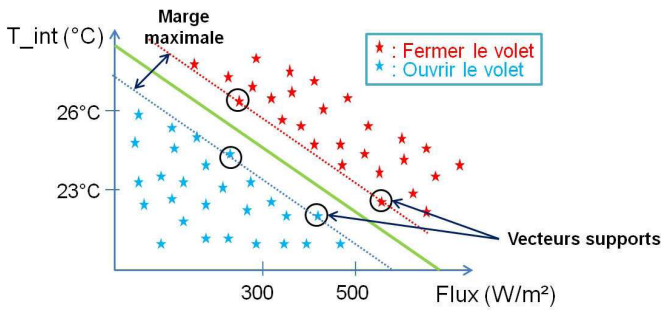

FIGURE *2.10: L'hyperplan séparateur obtenu avec le SVM. Le principe de fonctionnement du SVM se base sur les vecteurs supports et la marge maximale*

Ainsi, la recherche d'une marge maximale est équivalent à minimiser $\frac{1}{2}\|w\|^2$.
Enfin, nous obtenons le problème d'optimisation sous contraintes suivant :

$$\begin{array}{ll} \text{minimiser} & \frac{1}{2}\|w\|^2 \\ \text{s.t.} & y_i(x_i w + b) \geq +1 \end{array}$$

Le problème ci-dessus est également connu comme un problème sous contraintes quadratiques qui a été largement étudié dans la littérature [38, 39, 40].

La figure 2.10 montre l'hyperplan séparateur des deux décisions pour notre exemple de la gestion des volets.

Soft margin linear SVMs Dans le cas non séparable, un nouveau paramètre, appelé variable d'écart, est introduit et pénalise les points qui peinent à choisir une classe. Le problème d'optimisation est maintenant sous les contraintes suivantes :

$$\begin{array}{rcll} x_i w + b & \geq & +1 - \xi_i & \text{for } y_i = +1 \\ x_i w + b & \leq & -1 + \xi_i & \text{for } y_i = -1 \\ \xi_i & > & 0 & \forall i \end{array}$$

Pourtant, la marge a tendance à diminuer puisque l'hyperplan optimal devrait réduire au minimum les cas de mauvaise classification tandis qu'une marge maximale est recherchée. Un moyen très pratique de trouver le compromis de ces conditions est d'introduire une pénalité pour les erreurs de classification. Un petit changement est donné au critère initial à minimiser qui devient désormais $\frac{1}{2}\|w\|^2 + C(\sum_{i=1}^{k} \xi_i)$. Le paramètre C est utilisé pour augmenter ou diminuer la pénalité des mauvaises classifications.

Exemple La figure 2.10 montre un exemple pour discriminer deux décisions d'ouvrir et de fermer le volet en utilisant le SVM. Une droite est ainsi obtenue et permet de séparer au mieux les deux nuages de points. En effet, si un nouveau échantillon se trouve en dessus de la droite, le volet est fermé tandis que si elle est en dessous de la droite, la décision inverse sera prise.

Nous avons présenté trois méthodes phares pour un problème d'apprentissage automatique. A côté de leur capacité d'apprendre des caractéristiques d'une base de données pour prendre la décision à partir d'une nouvelle échantillon, elles sont aussi capables de classer les attributs selon leur importance, ce qui pourrait nous aider par exemple à trier les mesures les plus indispensables pour le problème d'apprentissage et minimiser ainsi le coût d'implémentation en réduisant le nombre de capteurs à mettre en place. Cela fait l'objet de présentation de la partie suivante.

2.4.5 Over-Fitting

Le problème de "surapprentissage" (ou "over fitting" en anglais) peut apparaître pendant l'application des algorithmes d'apprentissage. En effet, les résultats obtenus dans une situation de surapprentissage n'est pas capable de généraliser les caractéristiques des données. C'est pourquoi, il perd ses pouvoirs de prédictions sur de nouveaux jeu de données [41, 42, 43].

Pour éviter le problème de surapprentissage pendant le développement d'un arbre de décision, des techniques de pré-élagage et post-élagage peuvent être appliquées. Le premier consiste à arrêter le développement de l'arbre avant que tous les exemples de l'ensemble d'apprentissage soient bien classés. Le deuxième technique construit l'arbre complet et il enlève les nœuds qui ne sont pas nécessaires [44, 45].

Une méthode simple pour les techniques Boosting et particulièrement AdaBoost d'éviter le problème de surapprentissage est de minimiser le nombre d'itérations tout en garantissant le résultat de classification souhaité avec une erreur minimale de l'apprentissage [46].

Quant à la méthode SVM, nous décidons de nous intéresser seulement au cas de séparation linéaire. En effet, cette représentation favorise l'interprétation et la compréhension lorsqu'on s'en sert pour construire un contrôleur logique par la suite.

2.4.6 Feature selection

Dans l'apprentissage automatique, la sélection de l'attribut (Feature selection) est le processus de sélectionner un sous-ensemble d'attributs qui est pertinent dans la construction du modèle. En effet, la base qui contient de données collectées peut contenir des mesures redondantes qui n'aident pas à la décision. De plus, cette approche permet d'alléger le problème d'apprentissage, de raccourcir ainsi le temps de résolution du problème et d'éviter le problème de surapprentissage [47, 48, 49].

Il est important de noter que le Feature selection est particulièrement utile dans le cadre du processus d'analyse des données, car il montre quelles caractéristiques sont importantes pour la prédiction, et comment ces caractéristiques sont liées. En effet, dans notre cas étude, cela nous aide par exemple à bien choisir le nombre de capteurs strictement nécessaires pour un meilleur apprentissage et ainsi réduire le coût d'implémentation du produit sur terrain.

De plus, cette technique nous apporte d'autres fonctionnalités comme :

— Améliorer l'intelligibilité du modèle ;
— Réduire le temps de calcul du modèle ;
— Éviter le problème d'over fitting.

La plupart des algorithmes d'apprentissage machine sont conçus pour apprendre quels attributs sont les plus appropriés à utiliser pour prendre leurs décisions.

FIGURE *2.11: Interface de bienvenue de Weka*

Par exemple, la méthode d'arbre de décision choisit les attributs les plus prometteurs pour mettre sur l'arbre et ils devront en théorie ne jamais être inutiles pour l'apprentissage.

D'une autre façon, les attributs peuvent aussi être sélectionnés efficacement à l'aide du poids obtenus par la formation AdaBoost. En effet, plus le pondérant est grand, plus l'attribut associé à ce classifieur est utile.

Le SVM est aussi un bon outil pour bien sélectionner les attributs. En effet, les attributs peuvent être classés en fonction du paramètre w qui est, géométriquement, perpendiculaire à l'hyperplan de séparation. Ainsi, une valeur faible ce coefficient signifie que l'attribut correspondant est redondant et plus la valeur de w_i est grande, plus l'attribut associé est important.

2.4.7 Weka

Weka (Waikato Environment for Knowledge Analysis [50]) est un logiciel libre et gratuit, développé sous Java, à Université de Waikato en Nouvelle-Zélande. Il propose un ensemble d'outils permettant de :

— Manipuler et d'analyser des fichiers de données, en implémentant la plupart des algorithmes d'intelligence artificielle (supervisé ou non) du domaine de l'apprentissage automatique, parmi lesquelles il y a les arbres de decision, l'Adaboost et SVM
— Pré-traiter des données, analyser avec des fonctions de "Feature selection".
— Évaluer et visualiser les résultats.

La fenêtre au lancement de Weka est présentée ci-contre (Figure 2.11).

Tous nos futurs résultats représentés dans cette thèse sont obtenus avec Weka.

2.5 Environnement de simulation-Introduction

L'outil de simulation du bâtiment est un élément important lors du développement d'un contrôleur car il nous fournit l'évolution du comportement du bâtiment qui est soumis à différentes conditions externes ou internes. De plus, il nous permet de raccourcir le temps de développement des

stratégies de commande mises en place ainsi que d'évaluer leur performance, ce qui serait impossible en conditions réelles pour des raisons de coût d'investissement pour la construction des bâtiments ainsi que le temps nécessaire pour le test.

Plusieurs outils dédiés à la modélisation et la simulation des bâtiments existent dans la communauté, parmi lesquelles nous pouvons citer TRNSYS [51], EnergyPlus [52] ou SIMBAD [53] qui proposent tous des modèles relativement précis. Pourtant, TRNSYS et EnergyPlus sont plutôt dédiés à l'estimation de la consommation globale du bâtiment. Uniquement des simples contrôleurs de type PID ou TOR y sont actuellement proposés et ces outils ne sont pas adaptés, à notre avis, à la conception des contrôleurs avancés. C'est pourquoi nous avons décidé d'adopter SIMBAD pour nos travaux car il a pour avantage non négligeable de pouvoir être totalement intégré à l'environnement de Matlab-Simulink qui facilite la réalisation de l'optimisation de la commande prédictive ou encore de l'identification des modèles pour les bâtiments. De plus, il n'est pas nécessaire d'avoir recours à des modules pour pouvoir coupler un logiciel spécifique du bâtiment et l'environnement Matlab, comme cela peut-être le cas lorsqu'on utilise des outils tels qu'EnergyPlus ou TRNSYS.

2.5.1 Présentation SIMBAD

SIMBAD (SIMulator of Building And Devices) est une bibliothèque de composants Matlab/Simulink dédiée à la modélisation et à la simulation dynamique des bâtiments. Elle permet de développer et de tester des contrôleurs ou des stratégies de contrôle des équipements dans le bâtiment tels que des systèmes de chauffage, d'occultation ou de ventilation etc. L'utilisation de SIMBAD pour répondre à des problématiques d'optimisation de la gestion énergétique des bâtiments se trouve dans de nombreux travaux dans la littérature. Ainsi, on peut citer par exemple les travaux de Lefort [54] et de Morosan [55] qui développent et testent des contrôleurs prédictifs pour la gestion de l'énergie dans le bâtiment.

La description du bâtiment ainsi que ses systèmes dans SIMBAD est hiérarchisée et structurée grâce à un fichier XML qui permet de configurer l'ensemble des paramètres tels que :

- Les conditions climatiques extérieures (données météorologiques) ;
- L'occupation du bâtiment ;
- Les équipements actifs installés (chauffage, rafraîchissement, ventilation, éclairage) ;
- La structure (géométrique et thermique) du bâtiment.

Il est important de souligner qu'un bâtiment est souvent modélisé par l'approche zonale dans SIMBAD, ce qui nécessite une description complète des éléments du bâtiments tels que les murs, les fenêtres, les types d'isolation murs etc. A partir de ces informations, un modèle de type "boîte blanche" est prédéfini et représente l'inertie, l'infiltration du bâtiment, les pertes etc. La figure 2.12 montre comment la structure du bâtiment s'est organisée avec ces différents éléments.

2.5.2 Cadre de travail

Pour nos travaux, un bâtiment avec deux zones a été pris en compte. Le plan du bâtiment est donné par la figure 2.13. La zone Salon dispose une surface de 27 m^2 (67.5 m^3) avec 2 ouvertures de 1 m^2 chacune à l'est et à l'ouest. De plus, une grande ouverture de 6 m^2 se trouve au sud pour profiter de l'énergie solaire gratuite. La surface de l'autre zone Chambre est 24 m^2 (60 m^3) avec aussi 2 ouvertures de 1 m^2 chacune à l'est et à l'ouest.

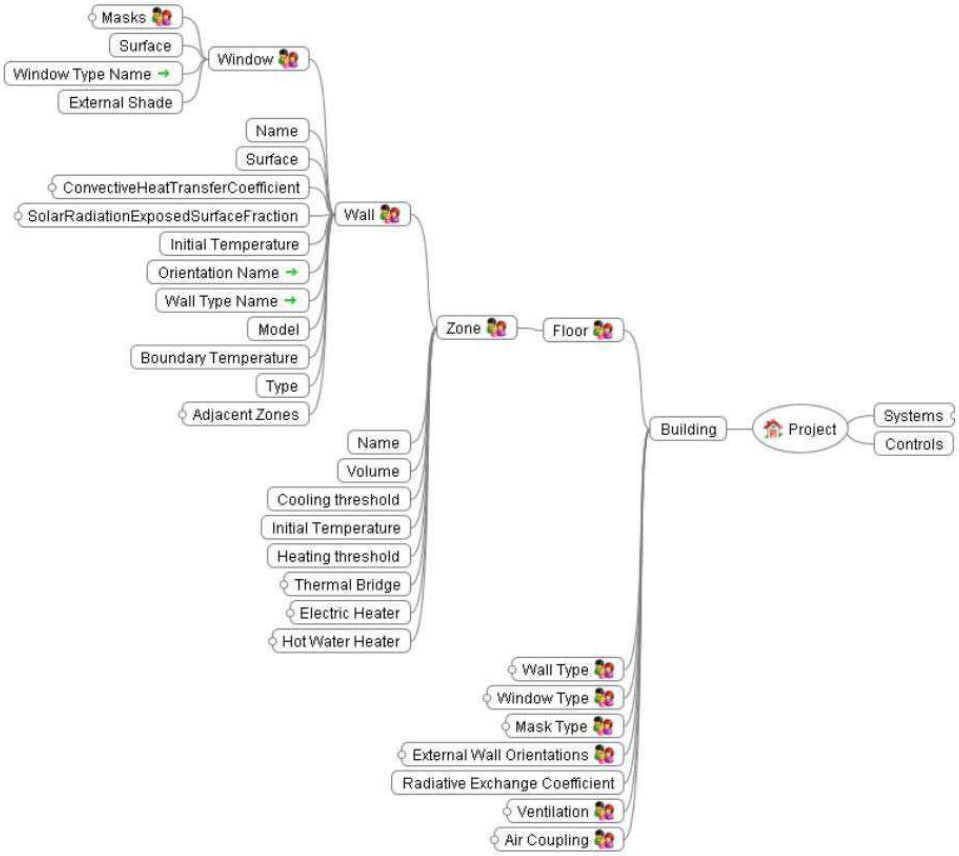

FIGURE *2.12: La description du bâtiment avec le fichier XML*

Les équipements dans la maison sont :

— Un volet pour chaque ouverture ;
— Un système de ventilation de type VMC simple flux ;
— Un radiateur électrique de 2kW dans chaque pièce ;
— Un système d'éclairage.

Nous utilisons des conditions météorologiques réelles qui ont été enregistrées à Vaulx-en-Velin (France) en 2011 [56] . En effet, cela nous permet de tester les performances des contrôleurs dans des conditions réelles, notamment en France où la température extérieure peut varier fortement d'un jour à l'autre (Figure 4.1).

2.6 Conclusion du chapite - Synthèse

Nos maisons ont subi des changements radicaux au niveau structurel ainsi qu'au niveau des équipements afin de mieux assurer les conforts pour les occupants. En effet, le design de nos logements a beaucoup changé en utilisant des matériaux plus performants en termes d'isolation thermique. De plus, ils sont munis de plus en plus d'équipements tels que le système de chauffage,

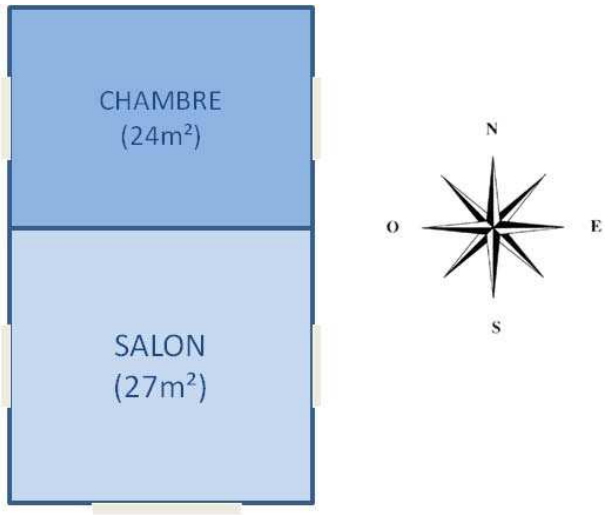

FIGURE *2.13: Plan de la maison*

de la ventilation, les occultations etc., qui sont aussi de moins en moins énergivores. L'évolution de la construction des maisons et de la performance des équipements entraîne aussi l'évolution des stratégies de contrôle et de pilotage de ces derniers.

De par la simplicité dans sa construction et la facilité dans son implémentation, le contrôleur logique reste le premier choix des ingénieurs de contrôle des bâtiments. Pourtant, il demande toujours une connaissance des experts dans ce domaine afin de mieux choisir des règles logiques ainsi que des seuils pour ces règles.

Face à ces difficultés, nous proposons ici une procédure pour construire un contrôleur logique en remplaçant le dit expert par un contrôleur optimal, puis nous obtenons des règles logiques et leurs seuils grâce à un apprentissage des comportements de ce contrôleur optimal. Notre procédure comporte donc deux étapes :

— Étape 1 : Construire un contrôleur optimal : nous avons choisi l'approche MPC avec l'avantage de pouvoir prendre en compte des différentes prédictions météorologiques ou de présence, qui sont particulièrement valorisants dans un problème de contrôle du bâtiment afin d'anticiper l'inertie de ce dernier. Le but de la construction du contrôleur MPC est de générer une base de données qui va servir pour l'étape 2 de la procédure.
— Étape 2 : Apprentissage : l'objectif de cette étape est donc d'apprendre des comportements du MPC, à partir desquels nous allons développer notre contrôleur logique. Nous mettons en œuvre des algorithmes dans le domaine d'apprentissage automatique basés sur la méthode de l'arbre de décision, Adaboost ou SVM.

Nous détaillons par la suite les différentes sous étapes dans la construction d'un contrôleur MPC ainsi que du problème d'apprentissage. Nous montrerons que les tâches ne sont pas simples et de nombreux défis doivent être relevés. En effet, les besoins pour construire un contrôleur optimal sont :

— Les modèles de prédiction - défi 1 : comment bien modéliser le bâtiment ? Plusieurs phénomènes complexes se produisent au sein du bâtiment, e.g. les échanges d'air ou de la

chaleur entre le milieu externe et interne du bâtiment. C'est pourquoi, modéliser un bâtiment n'est pas une tache évidente. De plus, les équipements possèdent souvent différents modes de fonctionnement qui apportent encore plus de difficultés à la modélisation.

— Les critères - défi 2 : comment trouver un compromis des critères ? En effet, nous devons faire face à un problème multicritère que nous transformons par la suite en un seul critère avec des pondérations. La question de trouver le compromis devient comment bien régler les pondération pour satisfaire aux mieux les différents conforts de l'occupant.
— Résolution du problème d'optimisation - défi 3 : comment faire face à la taille du problème qui est souvent de grande taille et qui nécessite un effort de calcul important ?

Après avoir réussi à développer ce contrôleur MPC, il faut ensuite mettre en œuvre des techniques du domaine "apprentissage automatique". Pour cela, nous avons besoin de :

— Construire une base de données - défi 4 : quelle est la durée de la simulation pour obtenir suffisamment de données qui permet de couvrir un maximum de cas possibles, sans se soucier du temps nécessaire pour la simulation ?
— Mettre en œuvre les algorithmes d'apprentissage - défi 5 : comment choisir l'algorithme le plus approprié ? Comment bien choisir les mesures utiles pour le problème d'apprentissage afin d'éviter le problème over-fitting ?
— Construire un contrôleur logique - défi 6 : comment interpréter les résultats obtenus grâce à des algorithmes d'apprentissage automatique ? Dans le cas d'un contrôle coordonné de plusieurs systèmes, faut-il privilégier un apprentissage global ou système par système ?

Notre méthodologie sera appliquée et sera présentée dans le chapitre suivant avec l'application sur le pilotage des occultations. Elle sera ensuite validée par une application plus compliquée qui propose de contrôler la ventilation en plus du pilotage des volets.

Deuxième partie

Premier cas d’étude simple : Gestion des occultations

CHAPITRE 3

Développement d'un contrôleur optimal avec la commande prédictive hybride HMPC

Sommaire

3.1 Contexte et motivation **35**
3.2 Développement de la commande prédictive hybride **36**
3.2.1 Modèles 37
3.2.1.1 Modèle de température intérieure du bâtiment 37
3.2.1.2 Modèle de la luminosité 38
3.2.2 Traduction mathématique des critères et des contraintes 38
3.2.2.1 Critère de confort thermique 39
3.2.2.2 Critère de confort visuel 40
3.2.2.3 Critère de confort sonore et de la durée de vie des volets 40
3.2.3 Problème d'optimisation 40
3.2.4 Résolution du problème 41
3.3 Résultats de simulation **42**
3.3.1 Conditions de simulation 43
3.3.2 Prévisions météorologiques parfaites 43
3.3.3 Prévisions météorologiques erronées 44
3.3.4 Comparaison avec un contrôleur classique 45
3.3.5 Économie sur la consommation énergétique 45
3.4 Conclusion **47**

3.1 Contexte et motivation

Dans ce chapitre, la première étape dans notre méthodologie sera présentée avec le développement d'un contrôleur optimal prédictif dans le cas simple du pilotage du système d'occultation. Les résultats de simulation obtenus nous fourniront ensuite une base de données pour l'étude de l'apprentissage du comportement ainsi que la construction d'un contrôleur qui est moins performant mais qui est plus simple à implémenter dans un produit commercial. Ces derniers points feront l'objet du chapitre suivant.

Le choix de contrôler les volets n'est pas un hasard. En effet, dans le but de réduire la facture énergétique des consommateurs, les bâtiments sont désormais équipés de grandes baies vitrées

et d'une isolation thermique performante. Ce nouveau concept permet de profiter du rayonnement solaire gratuit pour chauffer le bâtiment et donc de réduire la consommation en hiver. Pourtant, il entraîne de la surchauffe en été car la chaleur a du mal à s'échapper du bâtiment à cause de l'isolation renforcée. Un contrôle des occultations devrait maintenir le niveau de confort de l'occupant (thermique et visuel) en contrôlant le gain des apports solaire et garantissant un minimum de lumière naturelle sans faire appel au système d'éclairage électrique. Il contribue ainsi à une économie non négligeable sur le refroidissement et la consommation d'éclairage.

De nombreuses études ont été réalisées dans la littérature et elles ont montré un potentiel d'économies d'énergie en utilisant un contrôle automatisé des occultations [57, 58, 59]. De plus, l'économie d'énergie réalisée augmente considérablement si la stratégie de contrôle peut prendre en compte la prédiction du gain de chaleur et la prédiction de la température extérieure [10].

La suite de ce chapitre répond à la question de comment formuler un problème de commande prédictive. Nous commençons par le développement de différents éléments importants du problème : les modèles, la traduction mathématique des critères ainsi que la formulation du problème d'optimisation, qui est ensuite résolu avec la technique Branch and Bound. Enfin, différents résultats de simulation seront présentés à la dernière partie du chapitre pour évaluer la performance du contrôleur MPC en le faisant fonctionner sous différents scénarios et en comparant avec un contrôleur logique simple.

3.2 Développement de la commande prédictive hybride

Pour notre cas d'étude, notre stratégie de commande des volets ne se limite pas à la fermeture ou l'ouverture complète, elle possède en effet 2 positions de plus qui sont une fermeture à 50% et une fermeture à 75%. Ces deux positions intermédiaires correspondent respectivement à un bon niveau de lumière (i.e supérieure à 500 lux) et à un niveau minimal de luminosité dans la pièce (i.e 300 lux). Il convient à noter que ces choix de positions font partie des éléments du cahier des charges. Il est clair que nos variables de contrôle sont discrètes, $u_b \in \{0, 0,5, 0,75, 1\}$. D'une part, les deux positions intermédiaires nous donne plus de degré de liberté pour atteindre plus facilement nos objectifs de confort thermique sans détériorer le confort visuel des occupants. D'autre part, en considérant la position du store comme une variable discrète, il est plus facile de limiter la variation de position. En conséquence, il permet d'assurer le confort sonore des occupants et de garantir une meilleure durée de vie du moteur des volets.

C'est pourquoi nous nous sommes concentrés sur le contrôle du système hybride utilisant le MPC. La figure 3.1 présente le schéma de fonctionnement de notre contrôleur. En effet, nous y trouvons un modèle du bâtiment qui nous fournira des prédictions sur l'évolution de la température intérieure ainsi que du niveau de luminosité dans la pièce. De plus, notre stratégie de commande est capable de prendre en compte différentes prédictions dans ses calculs (i.e. les prédictions du flux solaire et de la température extérieure).

Il est important de noter que nous nous intéressons uniquement à piloter le grand volet au SUD du salon qui reçoit beaucoup plus d'apports énergétiques que les autres ouvertures.

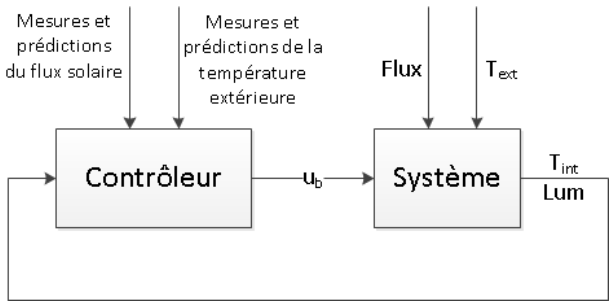

FIGURE *3.1: Le schéma du MPC pour la gestion des occultations*

3.2.1 Modèles

L'avantage principal de la commande prédictive MPC est sa capacité à anticiper le comportement futur du bâtiment sur une horizon de prédiction. Les modèles de prédictions jouent ainsi un rôle très important dans la construction du contrôleur. Il est clair que pour nos travaux, nous avons besoin d'un modèle de température intérieure ainsi que d'un modèle de niveau de luminosité dans la pièce afin de piloter notre volet afin d'assurer le confort de l'occupant. Chaque modèle est détaillé dans la suite de cette section.

3.2.1.1 Modèle de température intérieure du bâtiment

Le modèle du bâtiment SIMBAD est embarqué dans un schéma bloc SIMULINK et il est "fermé" aux utilisateurs. Par conséquent, le modèle mathématique n'est pas explicitement connu et il est identifié à partir d'une simulation. C'est pourquoi nous avons décidé d'utiliser le modèle de type boîte noire pour construire notre modèle de prédiction pour la température intérieure. Il s'agit d'un modèle de comportement global entrée-sortie qui peut être présenté, dans un cas discret, sous la forme suivante :

$$\begin{aligned} x(k+1) &= Ax(k)+Bu(k) \\ T_{int}(k) &= Cx(k) \end{aligned}$$

On obtient donc :

— $x(k)$ est l'état du système à temps k

— $T_{int}(k)$ est la sortie du système, i.e. la température intérieure

— $u(k) = \begin{bmatrix} \phi(k)(1-u_b(k)) \\ T_{ext}(k) \end{bmatrix}$ sont les entrées mesurables du système parmi lesquelles :

- T_{ext} est la température extérieure
- $\phi(k)(1-u_b(k))$ est l'énergie solaire reçue à travers l'ouverture, ce qui dépend clairement de l'intensité du flux solaire ϕ et la position du volet u_b. Nous avons choisi ici une

relation linéaire entre ces éléments afin de simplifier notre modèle.

Il est important de noter que le fait d'exprimer l'énergie reçue en fonction de l'ouverture du volet nous permet de réduire le nombre de modèles nécessaires à 4 modèles, qui correspondent aux 4 degrés d'ouverture du volet, à un seul modèle. De plus, nous supposons que la fermeture des volets ne contribue pas à renforcer l'isolation de la pièce. Par conséquent, la matrice A ne dépend pas de u. Les principaux facteurs qui influent sur la température intérieure du bâtiment sont le flux solaire et la température extérieure.

Identification Dans un processus de modélisation du système, l'identification des paramètres du système est l'une des étapes les plus importantes pour décrire le comportement du système avec des résultats précis.

Afin d'identifier le modèle de la température intérieure, nous avons effectué une simulation d'une durée de 2 mois avec SIMBAD. Nous avons ensuite sauvegardé les entrées et la sortie de température intérieure nécessaires pour l'identification. L'outil utilisé est le toolbox "Identification" de Matlab.

Pour ne pas influencer le comportement, le système de ventilation et le chauffage n'ont pas été contrôlés (i.e la ventilation tourne à la vitesse 1, ce qui est réglementé et le chauffage est éteint). Il faut noter aussi que nous avons fait le choix de déterminer les matrices d'états A, B, C une seule fois avec le volet ouvert, i.e. $u_b = 0$

Résultats Dans cette partie, nous montrons une comparaison de la courbe prédite (rouge) par le modèle obtenu et la courbe réelle (bleu) fournie par SIMBAD. La figure 3.2 présente le résultat obtenu par une prédiction à partir de 1 heure. Il est important de noter qu'un reconstructeur d'état a été aussi implémenté afin d'estimer l'état du système au début de la simulation.

Nous y voyons clairement que notre modèle a fourni une bonne prédiction de la température intérieure en fonction de la position du volet avec un horizon de prédiction qui est à 10 heures.

3.2.1.2 Modèle de la luminosité

La luminosité intérieure est aussi influencée par le flux solaire et la position du volet. Nous modélisons le niveau de luminosité dans la pièce avec un modèle linéaire simple dont l'équation est écrite comme ci-dessous :

$$Lum(k) = G\phi(k)(1 - u_b(k))$$

avec G est le gain de conversion entre le flux solaire $[W/m^2]$ et la luminosité $[lux]$.

3.2.2 Traduction mathématique des critères et des contraintes

Dans cette partie, les critères de confort ainsi que les contraintes de fonctionnement de l'équipement seront traduits pour être pris en compte dans la formulation du problème. En effet, les occultations sont pilotées pour éviter un apport solaire trop important qui pourrait causer une surchauffe, ce qui nuit au confort thermique de l'occupant. De plus, il ne faut pas fermer le volet tout le temps pour assurer un minimum de lumière naturelle dans la maison. Nous avons aussi une

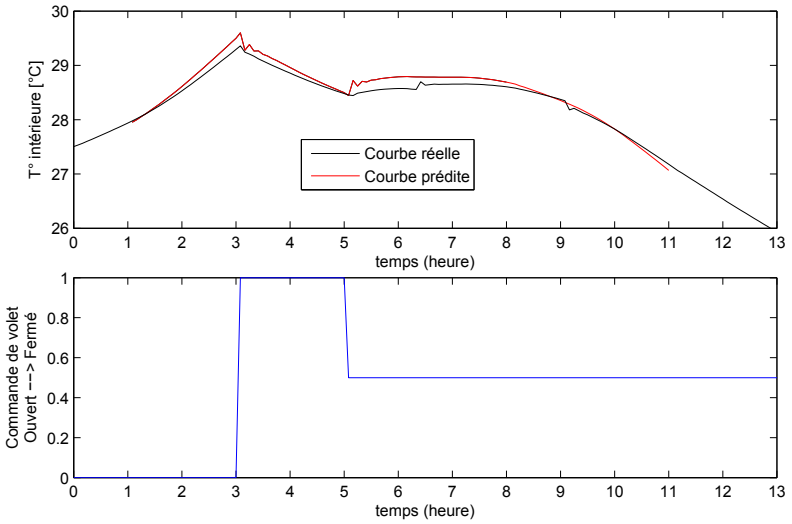

FIGURE *3.2: Performance du modèle thermique, résultats de comparaison entre la courbe réelle et la courbe prédite sur un horizon de prédiction de 10 heures*

contrainte liée au fonctionnement du volet, afin de maximiser sa durée de vie : il faut limiter ses mouvements.

3.2.2.1 Critère de confort thermique

Nous avons parlé de la complexité de la notion de confort thermique de l'être humain qui peut intégrer de nombreux paramètres, i.e. la température de l'air, la température des parois, l'habillement etc. En effet, une consigne à $24\,^{\circ}\mathrm{C}$ ou $25\,^{\circ}\mathrm{C}$ en période de chauffe n'est pas du tout satisfaisante pour les occupants. Le diagramme de Brager [60] (cf. Figure 3.3) propose une façon simple de prendre en compte cet aspect confort adaptatif selon la saison. Il permet d'obtenir une plage de confort en fonction de la température extérieure. Par exemple, quand il fait froid dehors (la température extérieure est inférieure à $5\,^{\circ}\mathrm{C}$), le confort est atteint pour une température intérieure qui se trouve en $17\,^{\circ}\mathrm{C}$ et $22\,^{\circ}\mathrm{C}$.

Pourtant, notre objectif donné est d'assurer le confort thermique de l'occupant, avec une température intérieure qui ne dépasse pas $26\,^{\circ}\mathrm{C}$. Les raisons pour cela sont multiples. Tout d'abord, nous nous intéressons uniquement à l'aspect protection des occultations contre les apports solaires, c'est pourquoi nous étudions seulement en période de mi-saison et en période de chauffe où une consigne à $26\,^{\circ}\mathrm{C}$ est acceptable. De plus, dans le soucis de simplicité pour le problème d'apprentissage qui suit derrière, nous nous contentons d'avoir une seule consigne. La consigne adaptative reste néanmoins dans notre esprit et nous traiterons aussi ce soucis dans le chapitre suivant.

Le critère J_{temp} sur la température intérieure peut être défini comme ci-dessous :

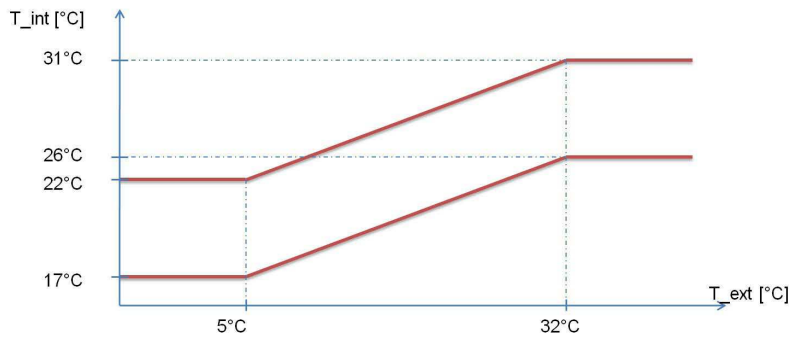

FIGURE *3.3: Diagramme de Brager*

$$J_{temp}(T_{int}) = \begin{cases} 0 & \text{if } T_{int} \leq 26 \\ T_{int} - 26 & \text{if } T_{int} > 26 \end{cases}$$

Nous pénalisons ainsi le critère du confort thermique quand ce dernier dépasse la consigne voulue qui est à $26\,^{\circ}\mathrm{C}$.

3.2.2.2 Critère de confort visuel

Nous essayons de maintenir un minimum de lumière du jour dans la pièce, par exemple, 300 lux dans la chambre pour le confort visuel quand cela est possible. Il est à noter que nous ne pilotons pas les systèmes d'éclairage.

Nous pénalisons ainsi le critère visuel lorsque le niveau de luminosité devient inférieur à 300 lux uniquement quand il fait jour.

$$J_{lum}(Lum) = \begin{cases} 0 & \text{if } Lum \geq 300 \\ 300 - Lum & \text{if } Lum < 300 \end{cases}$$

3.2.2.3 Critère de confort sonore et de la durée de vie des volets

Afin d'éviter l'inconfort sonore de l'utilisateur, il est nécessaire d'avoir un nombre limité de changement de position du volet. Un nombre limité de changement de position du volet pourrait même aider à prolonger la durée de vie de ce dernier.

Le critère suivant pénalise un changement de position du volet :

$$J_{mov}(\Delta u_b(k)) = |u_b(k) - u_b(k-1)|$$

3.2.3 Problème d'optimisation

Après avoir traduit tous les critères de confort ainsi que la contrainte de fonctionnement du volet, notre problème d'optimisation peut alors s'écrire comme suit :

$$\min_{U_b(k:k+N_h-1)} \sum_{j=1}^{N_h} [\gamma J_{mov}(\Delta u_b(k+j-1)) + \delta J_{lum}(Lum(k+j-1)) + \theta J_{temp}(T_{int}(k+j))] \quad (3.1)$$

s.t : $\forall j = 1 \ldots N_h$,

$$\begin{aligned} x(k+j+1) &= Ax(k+j) + B \begin{bmatrix} \phi(k+j)(1-u_b(k+j)) \\ T_{ext}(k+j) \end{bmatrix} \\ T_{int}(k+j) &= Cx(k+j) \\ Lum(k+j) &= G\phi(k+j)(1-u_b(k+j)) \\ u_b(k+j-1) &\in \{0, 0.5, 0.75, 1\} \end{aligned}$$

Les valeurs 0 et 1 de la position du volet correspondent respectivement à une ouverture et une fermeture complète de celui-ci tandis que 0.5 et 0.75 sont deux positions de fermeture intermédiaire, i.e. fermeture à 50% et à 75%.

Il est important de souligner que les paramètres γ, δ, θ dans l'équation (3.1) sont des pondérations sur les différents conforts et il est nécessaire de les régler judicieusement.

Il est clair qu'il existe une corrélation entre ces paramètres γ, δ, θ . En effet, la figure 3.4 montre l'évolution de la violation du confort thermique et visuel, ainsi que le nombre total de changements de position du volet pour une période de 10 jours. Nous varions la pondération sur le confort visuel δ en gardant les deux autres paramètres γ, θ.

Il est montré que la violation du confort visuel et le nombre de mouvements du volet diminuent lorsque la pondération sur le confort visuel augmente. Cependant, en ce qui concerne la violation de température, une tendance inverse a été observée car le contrôleur a eu tendance à laisser le volet ouvert et de ne pas le solliciter souvent afin d'assurer le confort visuel. Par conséquent, le confort thermique est gravement dégradé car il n'y a plus de protection contre les apports solaires.

C'est pourquoi, un bon compromis doit être trouvé afin d'équilibrer le confort thermique et visuel. Par exemple, la figure 3.4 montre pour les valeurs de γ et θ retenues qu'une valeur 8 du poids sur le confort visuel pour lesquels les courbes se croisent semble raisonnable. De plus, on obtient également une valeur moyenne du nombre de changements de position avec cette valeur.

3.2.4 Résolution du problème

La dernière étape pour le développement d'un contrôleur MPC est de résoudre le problème d'optimisation. Pour cela, nous adoptons la méthode de résolution Branch and Bound que nous avons présenté précédemment.

A chaque instant, les étapes suivantes sont effectuées :

— A chaque instant k, l'optimiseur obtient :
  - la prédiction de la température extérieure $T_{ext}(k : k+N_h-1)$,
  - le flux solaire $\phi(k : k+N_h-1)$,
  - la mémorisation de l'état de la position du volet $U_b(k-1)$,
  - l'état actuel du modèle $x(k)$.

— L'optimiseur calcule ensuite la meilleure séquence de position du volet en minimisant la fonction de coût J.

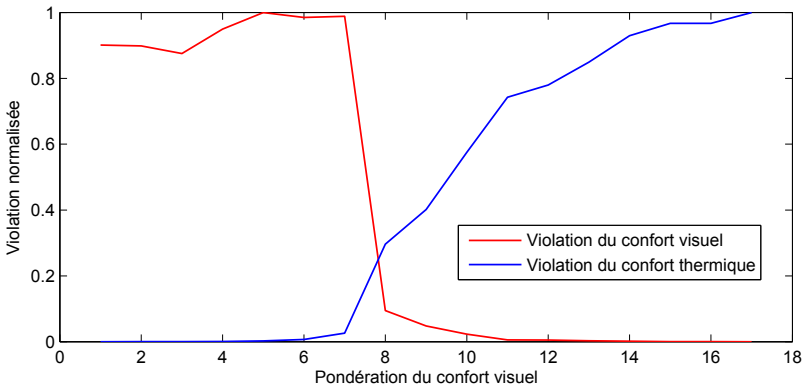

FIGURE *3.4: L'influence de la pondération sur les critères*

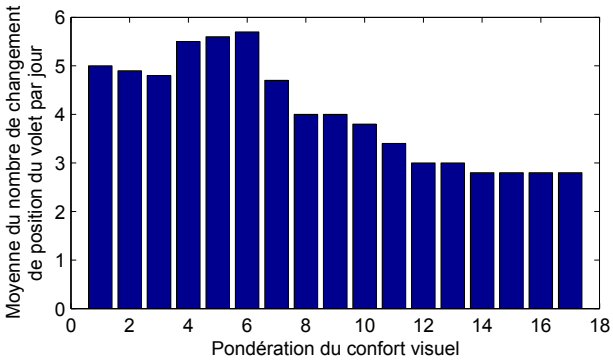

FIGURE *3.5: L'influence de la pondération sur le nombre de mouvements*

— Seul le premier élément de la séquence est appliquée, puis la procédure est répétée.

L'horizon de prédiction N_h est un autre paramètre qui doit être judicieusement choisi. Il est montré dans la figure 3.6 que la violation de la température varie très peu à partir d'une horizon de 3,5 heures, ce qui semble raisonnable d'un point de vue pratique. En outre, il permet de réduire considérablement les problèmes de calcul. De plus, avec notre méthode de recherche de solution optimale B & B, nous avons enregistré une moyenne de 2776 calculs à chaque pas de temps au lieu de 16384 (4^7). Il permet ainsi à réduire le temps de calcul d'environ 83 %. Il convient de noter aussi que notre modèle de prédiction de température intérieure ne prend pas en compte l'isolation du volet. C'est pourquoi, il est décidé de ne pas piloter le volet pendant la nuit afin de diminuer l'effort de calcul.

3.3 Résultats de simulation

Il est important de rappeler que l'objectif principal de ce chapitre est de développer un contrôleur optimal qui va servir ensuite pour le problème d'apprentissage. C'est pourquoi, notre but est

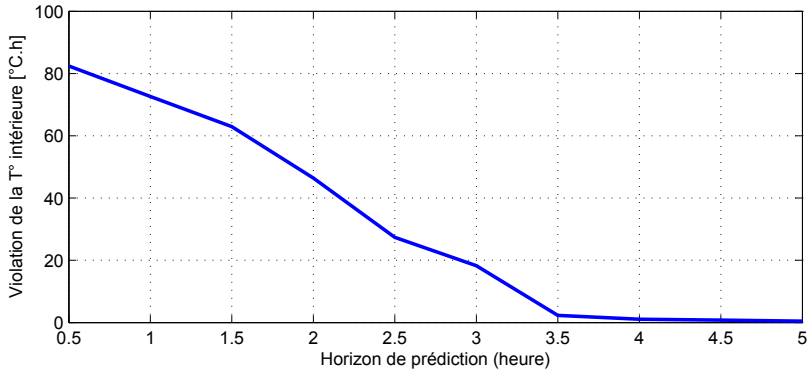

FIGURE *3.6: Relation entre la violation de la température et l'horizon de prédiction*

de présenter dans cette partie les résultats obtenus avec le MPC dans un cas parfait, i.e. toutes les prédictions sont connues parfaitement en avance. De plus, nous voulons aussi tester la robustesse du contrôleur obtenu dans le cas où les données de prévisions météorologiques sont manquantes. Enfin, une comparaison avec d'autres contrôleurs, par exemple un contrôleur proportionnel P ou un contrôleur simple basé sur l'habitude de l'occupant sera présenté afin de démontrer l'économie potentielle pour la facture énergétique.

3.3.1 Conditions de simulation

Les données météorologiques de SIMBAD sont en générale obtenues en moyennant les données de 20 ans de mesures. Nous avons décidé d'utiliser les conditions météorologiques réelles enregistrées à Vaulx-en-Velin (France) en 2011. Il nous permet de tester les performances MPC dans des conditions réelles. Nous simulons une semaine, à partir du 118e jour au 124e jour.

La figure 3.7 montre les conditions extérieures avec le flux solaire, la température extérieure ainsi que la température intérieure qui est obtenue avec aucune protection du volet, i.e. le volet est complètement ouvert. Nous pouvons observer un pic de température intérieure de $32\,^{\circ}\mathrm{C}$ enregistré dans la pièce, ce qui dégrade sérieusement le confort thermique de l'occupant.

Afin de réduire les problèmes liés au temps de calcul et d'assurer le confort sonore, la décision sur le changement de position du volet n'est prise qu'une seule fois toutes les 30 minutes.

3.3.2 Prévisions météorologiques parfaites

Comme mentionné précédemment, les conditions extérieures sont censées être parfaitement connues à l'avance dans ce cas.

La figure 3.8 montre les résultats de la simulation avec notre algorithme de commande prédictive. Nous pouvons constater que la température intérieure est maintenue à $26\,^{\circ}\mathrm{C}$ même quand l'énergie solaire est élevée et le volet est ouvert à la fin de la journée afin d'assurer le confort visuel des occupants. Le nombre de mouvements du volet est maintenue inférieur à 5 mouvements par jour, ce qui est largement acceptable pour l'occupant.

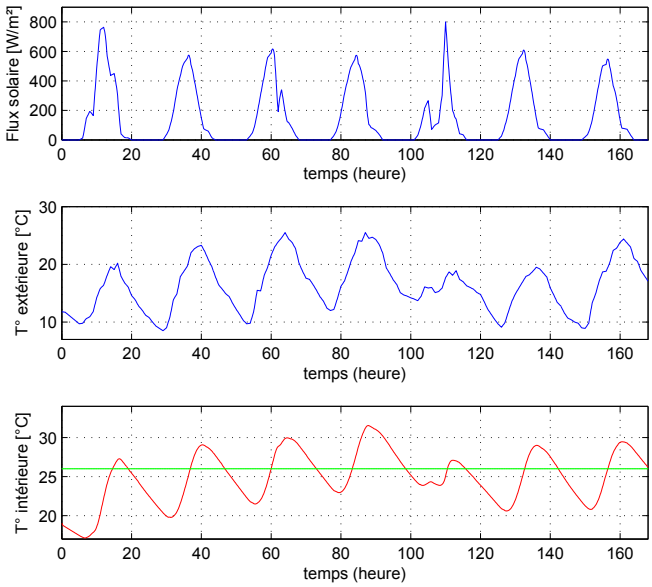

FIGURE *3.7: Conditions météorologiques avec les mesures de température extérieure et le flux solaire. Le confort thermique des occupants s'est dégradé avec la température intérieure qui dépasse le seuil de confort fixé à* $26\,^{\circ}\mathrm{C}$

3.3.3 Prévisions météorologiques erronées

Dans cette partie, nous supposons que les prévisions météorologiques ne sont pas connues à l'avance et que seules les mesures apportées de nos capteurs en temps réel sont à notre disposition. Afin de fournir les données de prédiction au MPC, nous avons décidé de garder la mesure venue des capteurs à une valeur constante sur tout l'horizon de prédiction. En conséquence, les prédictions fournies par le modèle devraient être différentes du comportement du bâtiment.

En dépit de cette difficulté, le MPC a bien régulé la température intérieure qui est maintenue autour de la consigne comme le montre la figure 3.9. En ce qui concerne le nombre de mouvements, nous pouvons observer qu'il y a seulement deux mouvements de plus pendant le deuxième et le sixième jour, par rapport aux résultats obtenus avec une prédiction parfaite. La raison est que le contrôleur MPC croit encore en une augmentation de la température dans la soirée et a décidé de fermer le volet tandis que l'énergie solaire a tendance à diminuer dans cette période de la journée. En outre, le MPC décide d'ouvrir plus tard le volet le soir que dans le cas précédent. Une performance acceptable du MPC en dépit d'une mauvaise prédiction météorologique peut s'expliquer par 30 minutes entre deux décisions sur le changement du volet. En effet, même si le contrôleur prédit moins d'apport solaire reçu, il réussit à rattraper son erreur au calcul suivant, i.e. 30 minutes plus tard. Pendant ce temps, la température monte ou baisse peu. Par conséquent, il est démontré que notre choix de 30 minutes semble approprié et justifié.

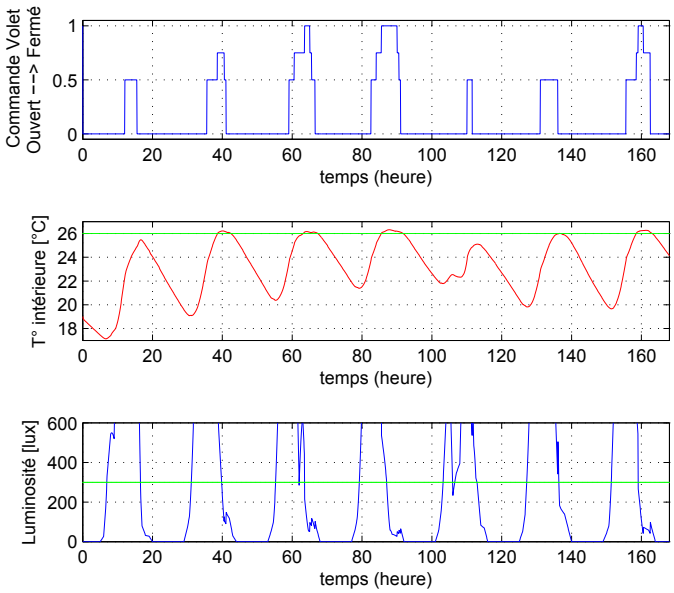

FIGURE *3.8: Résultats obtenus avec le MPC. Les conditions météorologiques sont parfaitement connues à l'avance.*

3.3.4 Comparaison avec un contrôleur classique

Dans cette section, nous mettons en œuvre le contrôleur classique P pour contrôler le volet. Il est clair que cette approche ne peut pas être utilisée dans la pratique. En effet, un bâtiment possède une inertie assez importante et le régulateur P est souvent en retard pour fermer le volet afin de lutter contre le flux solaire (cf. Figure 3.10). En conséquence, le régulateur P ne respecte pas la consigne, qui a été fixé à $26\,^{\circ}\mathrm{C}$ et les occupants se trouvent face à une situation d'inconfort.

Les volets ont aussi tendance à trop bouger et ils s'ouvrent en continu. En effet, nous pouvons remarquer par exemple au troisième ou au septième jour que le volet s'ouvre puis se ferme puisque la température oscille près de la consigne. Par conséquent, nous obtenons un cycle de vie des volets beaucoup plus court et les occupants se trouvent dans une situation d'inconfort sonore à cause de ce changement fréquent.

3.3.5 Économie sur la consommation énergétique

Afin d'étudier le potentiel d'économie énergétique obtenu avec le MPC, nous comparons ici notre contrôleur avec une stratégie simple adoptée par beaucoup d'occupants : ils ouvrent les volets le matin et ils les ferment le soir. En outre, un système de climatisation est également inclus dans le bâtiment avec une consigne à $26\,^{\circ}\mathrm{C}$ afin d'assurer le confort thermique. Nous effectuons une simulation pendant un mois d'été et nous comparons la consommation d'énergie dans les cas de figures suivants : le contrôle MPC avec une prévision météorologique parfaite, le contrôle MPC lorsque cette information est manquante (la température extérieure et le flux solaire manquants),

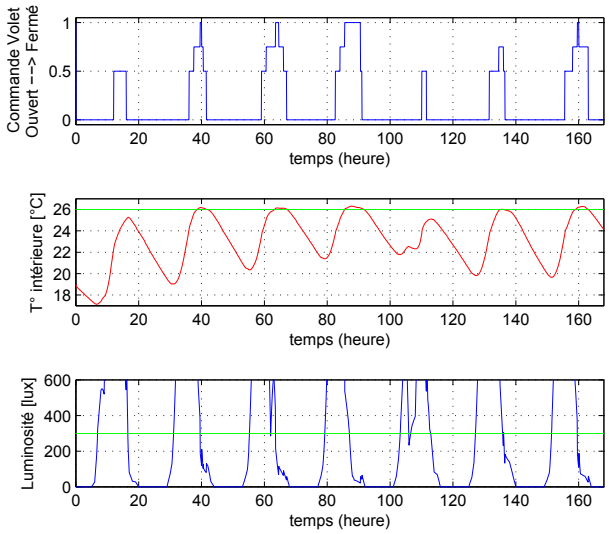

FIGURE *3.9: Résultats obtenus avec le MPC. Les conditions météorologiques ne sont pas connues à l'avance, elles sont bloquées sur tout l'horizon de prédiction à la valeur constante*

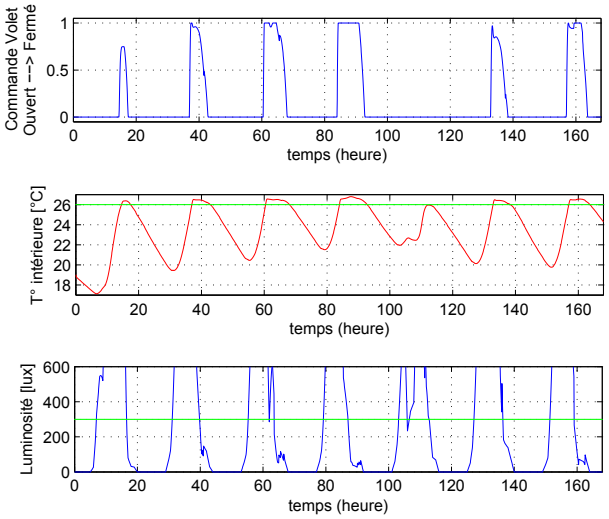

FIGURE *3.10: Résultats obtenus avec un contrôleur classique P*

	Contrôleur classique	Contrôleur P	MPC avec prédictions parfaites	MPC avec prédictions erronées
Consommation énergétique $[kWh]$	106	20	15	15
Mouvements du volet [par jour]	2	45	4.3	4.4
Violation de luminosité $[\times 10^3 lux.h]$	7.904	9.901	9.274	9.385

TABLE *3.1: Tableau de consommation de différents contrôleurs*

le contrôleur classique P et le contrôleur en fonction des comportements des occupants.

Il est démontré qu'en utilisant simplement deux mouvements du volet supplémentaires par jour, le contrôleur MPC permet de baisser considérablement la consommation d'énergie comme indiqué dans le tableau 3.1 alors que le contrôle classique atteint une violation de luminosité très proche du MPC avec les prédictions parfaites. Il faut souligner que la consommation est presque la même si les conditions extérieures sont connues ou pas à l'avance, mais le MPC fonctionne mieux dans le cas où les prévisions météo sont bien prédites, en particulier pour assurer le niveau de lumière.

3.4 Conclusion

Nous avons développé dans ce chapitre une stratégie de commande du système des occultations pour le confort de l'occupant, en particulier pour le confort thermique, le confort visuel et confort sonore. Le contrôle utilisant la commande prédictive MPC a montré des résultats satisfaisants, en anticipant l'évolution du comportement du bâtiment sur un horizon de prédiction, il est particulièrement efficace dans le cas où l'inertie du bâtiment est importante.

Le potentiel d'économie énergétique est considérable pour la période estivale. L'absence des prévisions météorologiques est également étudiée et nous montre qu'une performance satisfaisante du MPC pourrait obtenue malgré une manque de ces informations. En effet, nos études suggèrent que ces prévisions peuvent être créées en gardant les mesures comme une constante sur l'horizon de prévision. En outre, cela rendrait beaucoup plus simple la mise en œuvre du contrôle.

Nous avons relevé dans ce chapitre plusieurs défis concernant le développement du contrôleur optimal MPC. En effet, nous avons formulé un problème d'optimisation multicritères afin d'assurer différents conforts de l'occupant. Le choix des pondérations de ces critères sont aussi présentés avec notre méthode de réglage. Notre choix de l'outil Branch and Bound semble approprié pour résoudre notre problème d'optimisation. En effet, il permet de réduire considérablement le temps de simulation nécessaire.

Cependant, le contrôleur MPC demande toujours un microcontrôleur assez puissant pour pou-

voir piloter en temps réel, ce qui n'est pas évident à obtenir malgré nos suggestions de simplification dans la formulation du problème. C'est pourquoi nous nous sommes dirigés vers une méthodologie d'apprentissage des comportements du contrôleur optimal afin de construire des règles logiques à partir de ces connaissances. Cette idée et des résultats obtenus sont présentés dans le chapitre suivant.

CHAPITRE 4

Développement d'un contrôleur simple grâce à un apprentissage automatique

Sommaire

4.1 Motivation **49**
4.2 Formation d'une base de données **50**
4.3 Développement d'un contrôleur logique **50**
4.3.1 Apprentissage avec Arbre de décision 50
4.3.2 Apprentissage avec AdaBoost 50
4.3.3 Apprentissage avec SVM 50
4.3.4 Comparaison entre différentes méthodes 53
4.3.5 Etude avec SVM 54
4.3.5.1 Performance du contrôleur avec différents nombres d'entrées . . 55
4.3.5.2 Sensibilité des données 56
4.3.5.3 Adaptation des seuils en fonction du besoin de l'occupant 59
4.3.5.4 Simulation pendant une longue période 60
4.3.5.5 Portabilité de la solution 60
4.4 Conclusion **61**

4.1 Motivation

Nous avons montré dans le chapitre précédent que le MPC est un bon choix pour contrôler les volets grâce à sa capacité à prendre en compte différentes prédictions par rapport à d'autres stratégies classiques. Pourtant, le défi majeur de cette approche est sa mise en œuvre dans les produits industriels, qui ont une capacité de calcul limitée.

En conséquence, l'intérêt pour les méthodologies d'apprendre des comportements MPC, puis d'en déduire un contrôleur plus simple comme le contrôleur logique a connu une forte croissance au cours des dernières années. Les auteurs dans [61] ont proposé un modèle linéaire généralisé pour imiter les caractéristiques générales des résultats optimaux. Récemment, plusieurs algorithmes bien connus dans le domaine de l'apprentissage automatique tels que Support Vector Machine (SVM) et AdaBoost sont considérés dans [62]. Cependant, ces travaux n'ont porté que sur l'apprentissage des décisions binaires.

Dans ce chapitre, nous employons différentes méthodes présentées précédemment (cf. Section 2.4) afin de proposer un contrôleur plus simple à embarquer dans des produits commerciaux

en se basant sur des informations apprises à partir du contrôleur MPC. Nous allons étudier également l'importance de bien choisir les mesures prises en compte. En effet, non seulement cela permet de réduire la complexité du contrôleur logique mais c'est aussi nécessaire pour garantir une meilleure performance de la solution.

4.2 Formation d'une base de données

Afin d'obtenir suffisamment de données pour apprendre, nous effectuons une simulation pendant 30 jours en mars avec les mêmes conditions que le chapitre précédent. La figure 4.1 montre les conditions extérieures pendant ce mois. Des résultats obtenus pour une consigne à $24\,^{\circ}\mathrm{C}$ et à $26\,^{\circ}\mathrm{C}$ sont donnés ci-dessous (cf. figure 4.2). Nous effectuons un enregistrement tous les 5 minutes pour obtenir notre base de donnée.

Nous avons aussi montré précédemment que les données de prédictions ont peu d'importance sur la qualité de notre solution, en particulier pour la gestion des occultations. C'est pourquoi nous avons décidé d'enregistrer uniquement des données venant des capteurs, i.e. capteurs de température extérieure et intérieure, capteur de flux solaire et l'heure d'enregistrement. Par conséquent, nous obtenons quatre attributs pour notre problème d'apprentissage : la température intérieure T_{int} qui représente une image de l'état du système ainsi que le flux solaire $Flux$, la température extérieure T_{ext} et l'heure h.

4.3 Développement d'un contrôleur logique

Dans cette partie, l'ensemble de données enregistré précédemment est étudié avec nos trois méthodes dans le domaine Machine Learning. Des résultats d'apprentissage sont montrés et permettent de comparer l'efficacité de ces trois algorithmes.

4.3.1 Apprentissage avec Arbre de décision

Nous commençons avec la méthode de l'arbre de décision (C4.5). Avec Weka, nous avons la possibilité d'obtenir sous forme graphique pour le résultat de l'apprentissage (cf. figure 4.3).

4.3.2 Apprentissage avec AdaBoost

AdaBoost permet d'obtenir un résultat qui est censé être meilleur que l'arbre de décision en ajoutant une pondération à chaque seuil. Cependant, nous perdons l'avantage de la représentation graphique.

4.3.3 Apprentissage avec SVM

Dans notre cas d'étude, nous utilisons quatre positions du volet. En conséquence, nous devons faire face à un apprentissage SVM multiclasse. Dans la littérature, les stratégies les plus utilisées pour remédier à ce problème sont : méthode "one-versus-all" et méthode "one-versus-one [63]. La première stratégie tente de séparer les données appartenant à une classe de toutes les données

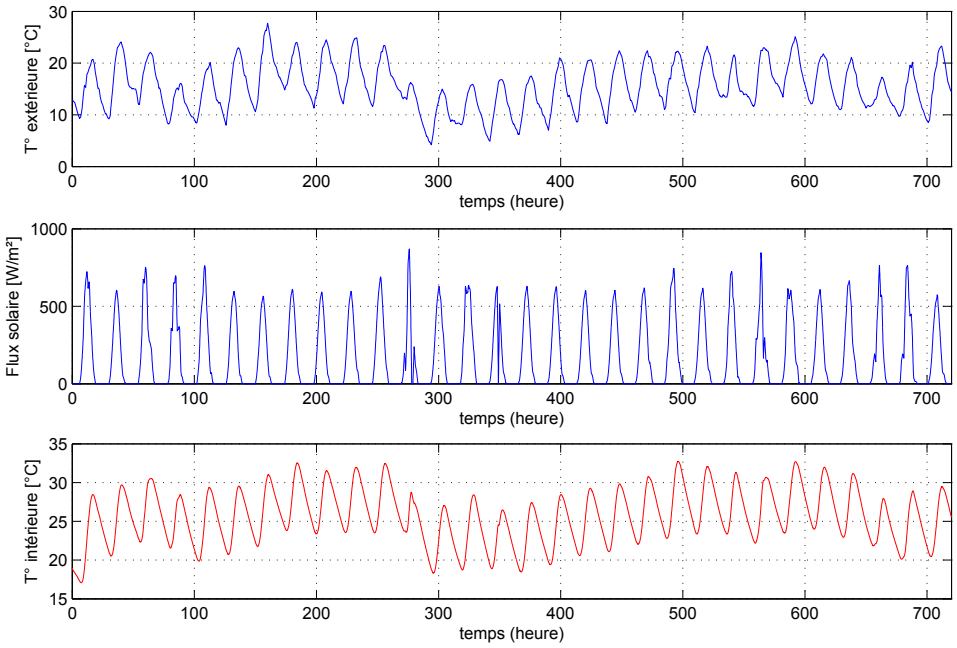

FIGURE *4.1: Conditions météorologiques en mars de la ville Vaulx-en-Velin (France) et la température intérieure sans protection des volets*

des autres classes tandis que la méthode d'un contre-un construit une SVM pour chaque paire de classes.

Cependant, nous pensons qu'il y a un moyen plus simple pour faire face à notre problème multiclasse. En effet, comme indiqué dans la formulation du problème en utilisant MPC hybride, nous limitons les mouvements intempestifs du volet. Ainsi, ce dernier ne se déplacera qu'à la position la plus proche de sa position actuelle si nécessaire. Une façon simple de résumer cette idée est de considérer le contrôleur comme une machine à états finis comme le montre la figure 4.4. Par exemple, si la position actuelle du volet est à moitié fermée alors le contrôleur ne prendra en compte que deux mouvements qui sont représentés dans la figure 4.4 par les vecteurs à traits solides. En effet, il va baisser le volet pour protéger le bâtiment contre une hausse de température intérieure ou il va l'ouvrir complètement pour profiter de la lumière naturelle ; si aucune de ces conditions est remplie, il restera dans sa position actuelle, i.e. 50% fermeture.

En conséquence, nous décidons de construire trois SVM pour trois paires de classes : la première hyperplan, noté $H_{(05_0)}$, sépare les données enregistrées pour une ouverture complète et des données pour une fermeture à moitié. La seconde hyperplan, noté $H_{(075_05)}$, divisera l'espace correspondant à la commande 50% fermeture et 75% fermeture. Le dernier hyperplan $H_{(1_075)}$, cherche à déterminer quand il faut fermer complètement ou fermer seulement à 75%. Nous avons ainsi simplifié considérablement le nombre d'hyperplans nécessaires, seulement 3 au lieu de 8

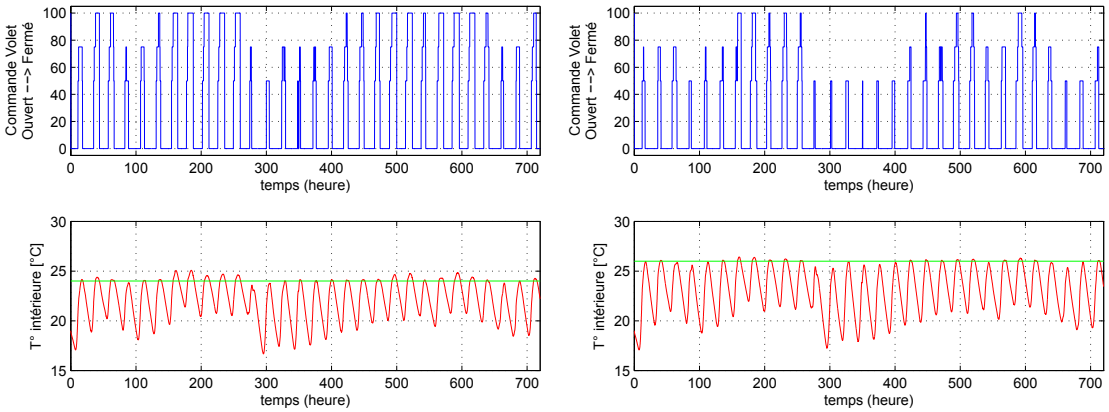

FIGURE *4.2: Résultats obtenus avec le MPC pour la consigne à* $24\,^{\circ}\mathrm{C}$ *et à* $26\,^{\circ}\mathrm{C}$

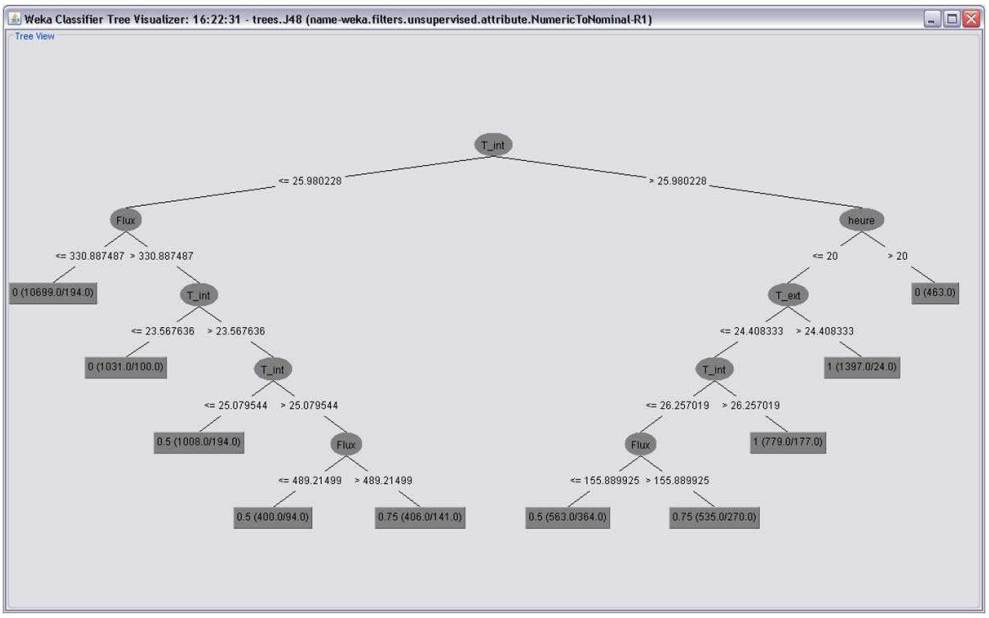

FIGURE *4.3: Résultat obtenu avec la méthode C4.5*

dans le cas où nous utilisons la stratégie "one-versus-one" tandis que nous restons toujours conformes à l'esprit de la commande optimale. En outre, nous pouvons éviter les cas où les données ne sont pas séparables linéairement, ce qui apparaît si la méthode "one-versus-all" est mise en place.

La forme générale d'un hyperplan peut être définie comme :

$$(H)\ : w^T x + b$$

où :

— $x^T = [T_{int}\ \ Flux\ \ T_{ext}\ \ h]$

— w, b sont les paramètres appris avec SVMs

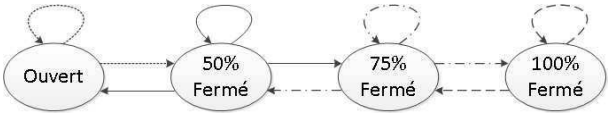

FIGURE *4.4: Présentation du contrôleur simplifié SVM sous forme de machine d'état*

4.3.4 Comparaison entre différentes méthodes

Dans cette partie, nous mettons en place trois contrôleurs simplifiés issus de l'apprentissage par les trois méthodes (C4.5-Arbre de décision, Adaboost et SVM) dont la construction a été présentée dans la partie précédente. La figure 4.5 présente les résultats obtenus avec les mêmes conditions météorologiques que pour les simulations avec le contrôleur MPC (i.e. pendant une semaine, du 118e au 124e, cf. Figure 3.6). Certes, leur performance est un peu moins bonne que celle du MPC, mais nous pouvons constater que la température est toujours bien régulée, seul un léger dépassement de la consigne au quatrième jour (le jour où l'énergie solaire est la plus importante) peut être constaté.

A notre avis, les trois méthodes peuvent apporter une solution qui est facilement implémentée dans un micro-contrôleur simple ayant une capacité de calcul limitée car ces contrôleurs simplifiés sont tous basés sur des règles très simples.

Quant à l'interprétabilité, l'arbre de décision nous fournit toujours une expression la plus simple à interpréter et à comprendre. Il est important de souligner aussi qu'il faut trouver une arbre de taille raisonnable afin de ne pas se perdre dans les règles et de confronter ainsi au problème "Over fitting". La méthode SVM avec notre idée de considérer le contrôleur sous forme d'une machine d'état semble être simple à interpréter et à comprendre aussi. La méthode AdaBoost nous semble être moins évidente à comprendre malgré son raisonnement par différents seuils.

Quant au codage, la méthode de l'arbre de décision et celle d'Adaboost demandent plus d'effort de codage que SVM. Notre proposition d'une machine d'état pour le SVM simplifie énormément le travail des programmeurs ainsi que les mises à jour de l'algorithme. En effet, la solution recherchée doit être non seulement facile à implémenter mais aussi adaptable à différentes régions ou différents climats. C'est pourquoi, les règles apprises pourraient être différentes selon ces conditions. Avec le SVM, il suffit de mettre à jour les paramètre de l'hyperplan, sans se préoccuper de la structure de machine d'état tandis que la méthode d'Adaboost et l'arbre de décision nécessitent un recodage total de l'algorithme implémenté dans le produit. De plus, le codage pour Adaboost devient de plus en plus fastidieux si nous voulons un meilleur résultat en augmentant le nombre d'itérations. Cela complique aussi l'interprétation de la méthode sachant que le nombre de paramètres est proportionnel au nombre d'itérations.

Une évaluation des trois méthodes selon trois critères (i.e. interprétabilité, implémentabilité et codage) est présentée dans le tableau 4.6.

Nous avons décidé d'utiliser l'outil SVM pour nos études plus poussées qui sont présentées par la suite.

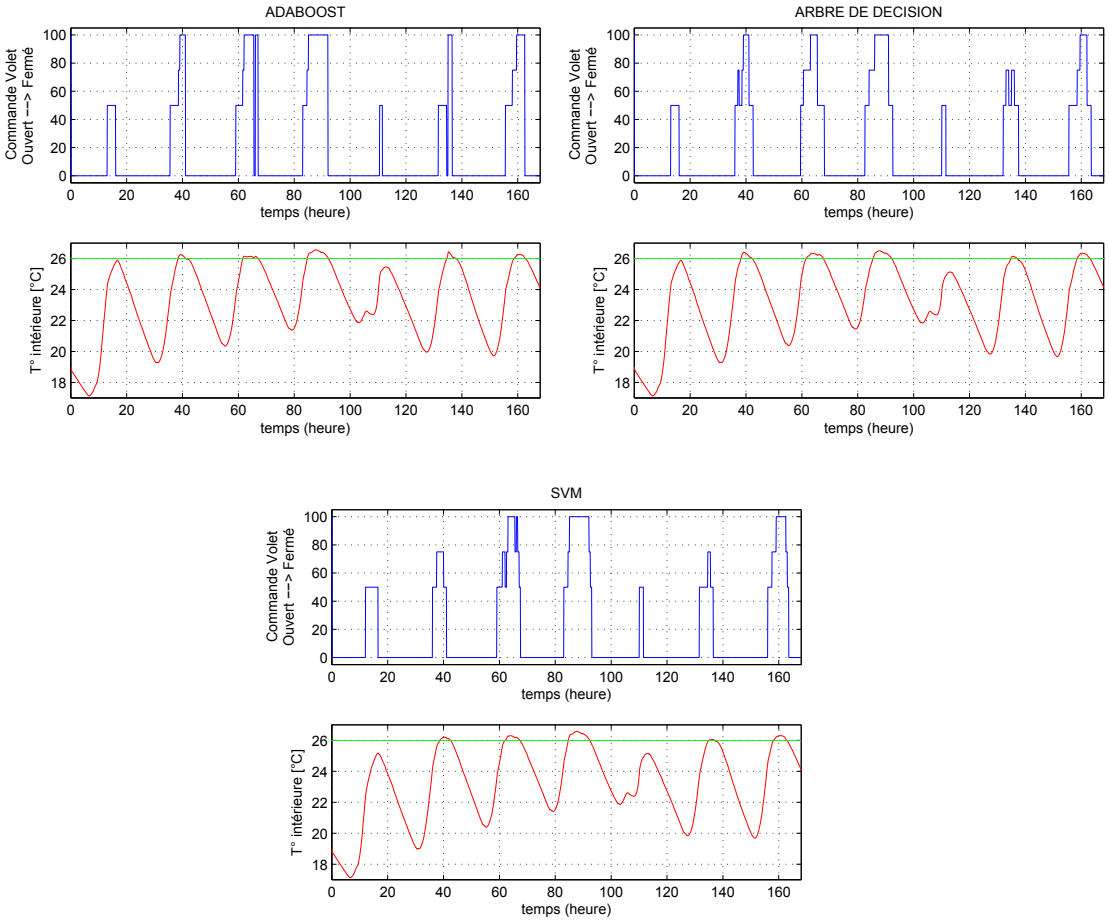

FIGURE *4.5: Résultats de simulation obtenus avec les différents contrôleurs simplifiés*

	Interprétabilité	Implémentabilité	Codage
Arbre de décision	★★★★★	★★★★★	★★★★☆
SVM	★★★★☆	★★★★★	★★★★★
AdaBoost	★★★☆☆	★★★★★	★★★☆☆

FIGURE *4.6: Comparaison de trois méthodes selon différents critères*

4.3.5 Etude avec SVM

Dans cette partie, une étude plus poussée est présentée avec l'algorithme SVM pour étudier l'importance d'un bon choix du nombre des entrées à prendre en compte. En effet, nous montrerons la performance de notre contrôleur appris avec SVM en fonction du nombre d'entrées selon leur classement fourni par la méthode "Feature selection". Ensuite, la sensibilité des résultats d'apprentissage est étudiée avec des données générées par différentes périodes de la simulation obtenues avec le contrôleur MPC. De plus, nous montrons que SVM propose une méthode simple

	T_{int}	Flux	T_{ext}	Heure
$H_{(1_075)}$	27.9	0.7	2.9	0.3
$H_{(075_05)}$	11.6	6.7	2	-1
$H_{(0_05)}$	28.3	7.9	7.4	0.1

TABLE *4.1: La valeur du paramètre w des hyperplans $H_{(1_075)}$, $H_{(075_05)}$, $H_{(05_0)}$*

pour adapter les paramètres du contrôleur logique aux besoins de l'occupant, ce qui ne pourrait pas obtenir avec Adaboost ou l'Arbre de décision. La portabilité de notre solution sera présentée dans la dernière partie.

4.3.5.1 Performance du contrôleur avec différents nombres d'entrées

Dans cette section, nous cherchons à comparer les performances du contrôleur optimal avec trois contrôleurs simplifiés, notés LC4, LC3 et LC2. Le premier contrôleur LC4 a été construit avec toutes les données d'entrée tandis que le contrôleur de LC3 et LC2 ont été appris avec respectivement trois entrées et deux entrées. Ce choix du nombre d'entrées à prendre en compte est basé sur le classement de l'importance de l'entrée grâce à l'algorithme SVM ("Feature selection", cf. Section 2.4.6). En effet, le coefficient du paramètre w représente le mérite de l'attribut dans le problème de l'apprentissage (cf. tableau 4.1). Il convient de souligner que les résultats sont obtenus avec des données qui sont normalisées sur l'échelle de 0 à 1 afin de comparer différentes entrées. La température intérieure semble être la mesure la plus importante vu que son coefficient w domine les autres, en particulier dans le cas de $H_{(1_075)}$ et $H_{(0_05)}$. Nous avons décidé de classer le gain solaire et la température extérieure à la deuxième et troisième position essentiellement en fonction de leur coefficient de $H_{(075_05)}$. L'entrée "temps" est cependant situé à la dernière place du classement. Ainsi, LC2 a été formé avec uniquement la température intérieure et les mesures de flux et les données de température extérieure ont été ajoutées au problème d'apprentissage pour LC3.

Nous discutons par la suite de la performance des différents contrôleurs en fonction de plusieurs critères. Le critère du confort thermique et du confort visuel mesure la violation de la contrainte en degrés Celsius-heure par jour et Lux-heure par jour. La valeur maximale de la température intérieure pendant la simulation et la moyenne des températures maximales quotidiennes sont également comparées. La contrainte sur le confort sonore est évaluée par le nombre de positions par jour ainsi que le nombre de cycles par jour. Il est à noter que ce dernier critère est le principal facteur qui influe le cycle de vie du moteur et il compte le nombre de fois où le volet change de direction.

Le tableau 4.2 montre les résultats de simulation en mars, effectuées par les contrôleurs simplifiés qui ont été appris grâce aux données générées en mars avec le contrôleur MPC optimal. Nous constatons que tous les contrôleurs d'apprentissage accomplissent assez bien leur travail et leur performance est très proche de la commande optimale. Il faut aussi souligner l'utilité de la technique "Feature selection". En effet, l'entrée «temps» semble être redondante parce que son coefficient d'hyperplan est très faible par rapport aux autres. Cela montre que l'ajout de données inutiles pour le problème de l'apprentissage va détériorer la performance du contrôleur appris.

Consigne : $26\,^{\circ}\mathrm{C}$	MPC	LC4	LC3	LC2
Violation thermique [$^{\circ}$C.h/jour]	0.26	0.37	0.35	0.39
Température maximale [$^{\circ}$C]	26.42	26.45	26.45	26.56
Moyenne de température maximale [$^{\circ}$C]	25.96	25.95	25.93	25.88
Violation visuelle [Lux.h/jour]	1800	1820	1813	1833
Nombre de mouvement [par jour]	4	4.4	4.3	4.8
Cycle [par jour]	2.1	2.17	2	2.2

TABLE *4.2: Comparaison de la performance de différents contrôleurs avec la consigne à* $26\,^{\circ}\mathrm{C}$ *en mars*

Ainsi, le résultat obtenu avec LC3 est le plus proche du contrôleur optimale que celui obtenu avec LC4.

Puisque la performance de LC2 est encore satisfaisante par rapport à la MPC optimale, nous décidons d'utiliser seulement deux entrées pour construire le contrôleur logique pour nos simulations expérimentales suivantes pour deux raisons. La première raison consiste à réduire le coût de déploiement pour l'industrie du bâtiment parce qu'un capteur pour la température extérieure n'est pas nécessaire. De plus, cela nous permet d'évaluer la robustesse du contrôleur approximé en effectuant les simulations avec le contrôleur qui est le plus simple et qui est aussi moins performant que les autres.

Par conséquent, les équations ci-dessus peuvent être réécrites dans une forme explicite comme suit :

$$\begin{aligned} L_{(05_0)} &: T_{indoor} = -\alpha_{(05_0)} Flux + \beta_{(05_0)} \\ L_{(075_05)} &: T_{indoor} = -\alpha_{(075_05)} Flux + \beta_{(075_05)} \\ L_{(1_075)} &: T_{indoor} = -\alpha_{(1_075)} Flux + \beta_{(1_075)} \end{aligned}$$

où :

— α est la pente de la droite

— β est la température de seuil

La figure 4.7 montre un exemple de la droite séparation $L_{(075_05)}$ en 2D avec la température intérieure et le flux solaire obtenus à partir d'une simulation en mars.

4.3.5.2 Sensibilité des données

Dans cette section, un problème d'apprentissage est réalisé avec des données générées par une simulation avec le MPC pendant différentes périodes de l'année, i.e. chaque mois de mars à septembre et une année entière. Les mois hivernaux ne sont pas pris en compte car les volets ne sont pas sollicités pendant cette période. Les résultats obtenus avec le SVM pour chaque paire de classes sont évalués en comparant la pente et le seuil de température des différentes simulations (Tableau 4.3, 4.4).

Les résultats montrent que ces paramètres appris à partir de différentes périodes sont dans les mêmes ordres de grandeur, sauf juillet. Nous concluons que les données générées par une période de mi-saison, à savoir en mars, peuvent être utilisés pour construire un contrôleur simplifié au lieu de prendre les données d'une année entière. Non seulement cela permet de réduire le

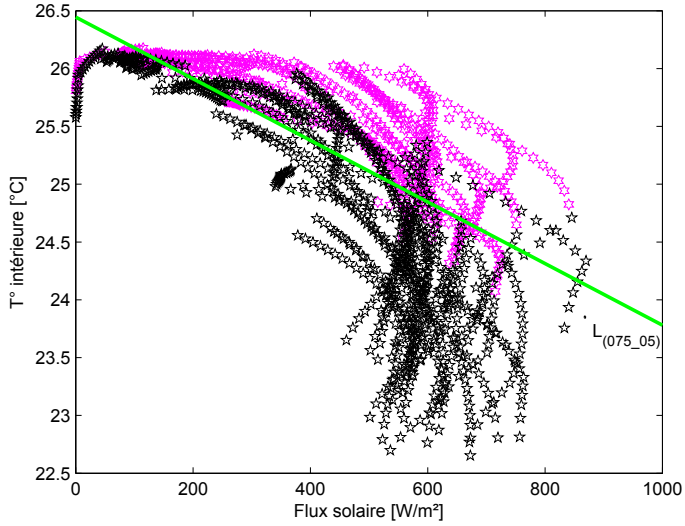

FIGURE *4.7: Exemple de $L_{(075_05)}$ obtenu par SVM. La décision est basée sur des mesures du flux solaire et de la température extérieure*

Paramètre α	MAR	AVR	MAI	JUIN	JUIL	AOUT	SEPT	1 année
$L_{(05_0)}$ $[10^{-3}]$	7	6.6	6	8.6	8.9	6.4	5.9	7.2
$L_{(075_05)}$ $[10^{-3}]$	2.7	2.1	1.9	2.2	1.4	2.8	3.1	2
$L_{(1_075)}$ $[10^{-3}]$	48	49	47	51	37	53	41	44

TABLE *4.3: Comparaison de la pente α obtenue par SVM selon la période d'apprentissage*

temps nécessaire pour acquérir des données, en particulier dans le cas de mesures réelles, mais le problème d'apprentissage est traité beaucoup plus rapidement car il y a moins de données.

En outre, différents comportements peuvent être constatés avec le contrôleur approximatif qui est appris avec les données de juillet, nommé LC2-J. En effet, il a tendance à fermer le store beaucoup plus tôt le matin et de l'ouvrir bien plus tard le soir par rapport au contrôleur appris dans Mars, noté LC2-M. Par conséquent, le paramètre $\alpha_{(05_0)}$ du LC2-J est plus grand que celui du LC2-M tandis que le seuil de température $\beta_{(075_05)}$ est plus petit(Tableau 4.5).

Afin d'évaluer les éventuelles conséquences si un contrôleur plus protecteur est employé à la mi-saison et inversement, si un contrôleur appris de mars est employé en juillet, nous effectuons

Paramètre β	MAR	AVR	MAI	JUIN	JUIL	AOUT	SEPT	1 année
$L_{(05_0)}$	27.1	26.8	26.5	27.6	27.6	26.7	26.7	27
$L_{(075_05)}$	26.4	26.4	26.1	26.2	25.9	26.4	26.1	26.2
$L_{(1_075)}$	26.2	26.2	26.1	26.1	26.1	26.1	26.1	26.1

TABLE *4.4: Comparaison du paramètre de seuil β obtenu par SVM selon la période d'apprentissage*

Résultats SVM	LC2-M		LC2-J	
	Pente α	β	Pente α	β
$L_{(05_0)}$	0.007	27.1	0.0089	27.6
$L_{(075_05)}$	0.0027	26.4	0.0014	25.9
$L_{(1_075)}$	$4.8\ 10^{-4}$	26.2	$3.69\ 10^{-4}$	26.1

TABLE *4.5: Comparaison de la pente α et le paramètre β des résultats obtenus par un apprentissage en mars et en juillet*

Consigne : $26\,^{\circ}\mathrm{C}$ en mars	LC2-M	LC2-J
Violation thermique [$^{\circ}$C.h/day]	0.39	0.22
Température maximale [$^{\circ}$C]	26.6	26.5
Moyenne de température maximale [$^{\circ}$C]	25.9	25.8
Violation visuelle [Lux.h/jour]	1833	1848
Nombre de mouvement [par jour]	4.8	4.4
Cycle [par jour]	2.2	2.03

TABLE *4.6: Comparaison de LC2-M et LC2-J selon différents critères avec une simulation en mars pour une consigne à $26\,^{\circ}\mathrm{C}$*

deux simulations en mars et juillet avec LC2-M et LC2-J. Les résultats sont présentés dans Tableau 4.6 and 4.7.

On peut remarquer que les volets restent souvent en position fermée lors de la simulation avec LC2-J. Par conséquent, la violation de la température est réduite tandis que la violation de la lumière augmente. En outre, ce dernier contrôleur tend à faire moins de mouvements que LC2-M. Cependant, les performances des deux contrôleurs sont encore très proches et le confort des occupants ne sont pas endommagé.

Une stratégie apprise en mars semble être suffisante pour le contrôle des volets, en particulier parce que la performance de LC2-M est encore acceptable à la mi-saison et même en été. Non seulement elle contribue à réduire le coût de l'enregistrement des données et la formation du problème mais aussi le coût de sa mise en œuvre où le contrôleur n'a pas besoin de changer sa stratégie tous les mois.

Consigne : $26\,^{\circ}\mathrm{C}$ en juillet	LC2-M	LC2-J
Violation thermique [$^{\circ}$C.h/jour]	16.4	15.8
Température maximale [$^{\circ}$C]	30.2	30.2
Moyenne de température maximale [$^{\circ}$C]	27.5	27.4
Violation visuelle [Lux.h/jour]	2481	2504
Nombre de mouvement [par jour]	5	4.7
Cycle [par jour]	2	1.8

TABLE *4.7: Comparaison de LC2-M et LC2-J selon différents critères avec une simulation en juillet pour une consigne à $26\,^{\circ}\mathrm{C}$*

Résultats SVM	$26\,^{\circ}\mathrm{C}$ en mars		$24\,^{\circ}\mathrm{C}$ en mars	
	Pente α	β	Pente α	β
$L_{(05_0)}$	0.007	27.1	0.0065	24.9
$L_{(075_05)}$	0.0027	26.4	0.0031	24.6
$L_{(1_075)}$	$4.8\ 10^{-4}$	26.2	$5.3\ 10^{-4}$	24.2

TABLE *4.8: Comparaison de la valeur de la pente α et du coefficient β obtenus grâce au SVM pour la consigne à $26\,^{\circ}\mathrm{C}$ et celle à $24\,^{\circ}\mathrm{C}$*

4.3.5.3 Adaptation des seuils en fonction du besoin de l'occupant

Une autre grande préoccupation pour la mise en œuvre du contrôleur dans un produit industriel est de s'adapter au besoin de l'occupant. En effet, ce dernier veut absolument garder la main sur le contrôleur qui devrait laisser ajuster ses paramètres par les utilisateurs. Dans cette section, nous proposons une méthodologie pour adapter le confort des occupants, soit en favorisant la lumière naturelle dans la maison, soit en privilégiant la température intérieure plus confortable. En effet, comme mentionné précédemment, il existe une corrélation directe entre le critère de température intérieure et le critère de luminosité. Ainsi, plus de lumière naturelle peut être obtenu en augmentant simplement la consigne de température.

En revanche, il est impossible d'effectuer un apprentissage pour chaque consigne de température et d'implémenter toutes ces stratégies de contrôle dans un produit industriel. Nous visons à proposer par la suite une méthode très simple et efficace pour répondre à ce problème au lieu de réapprendre à chaque changement de consigne.

Premièrement, nous montrons la comparaison de la pente et le seuil de température du contrôleur pour la consigne à $26\,^{\circ}\mathrm{C}$ et ceux pour la consigne à $24\,^{\circ}\mathrm{C}$ (Tableau 4.8). Ces paramètres sont obtenus avec des données générées par une simulation en mars.

Il est important de noter que les pentes des hyperplans sont dans les mêmes ordres de grandeur et qu'il existe une différence de $2\,^{\circ}\mathrm{C}$ entre deux seuils de température, ce qui est lié à la différence de consigne. En conséquence, nous proposons une nouvelle méthode pour construire un contrôleur pour une nouvelle consigne sans réapprendre avec le SVM en déplaçant simplement le seuil de température. En effet, les paramètres de bases sont appris pour la consigne de $26\,^{\circ}\mathrm{C}$ et quand l'occupant veut augmenter ou diminuer la consigne, le contrôleur peut s'adapter en ajustant tout simplement le seuil de température β en fonction de la différence entre la nouvelle consigne et 26.

Afin d'évaluer cette méthode, nous effectuons diverses simulations en mars avec deux contrôleurs. Le premier, désigné par LC2-M24, est le contrôleur approximatif qui est directement appris à partir des données du MPC en mars pour la consigne de $24\,^{\circ}\mathrm{C}$. L'autre, notée DLC-M24, est construit à partir de notre méthode tel que décrit ci-dessus, i.e. le seuil de température β est $2\,^{\circ}\mathrm{C}$ inférieur aux seuils de base qui ont été appris avec le MPC pour la consigne à $26\,^{\circ}\mathrm{C}$.

Le tableau 4.9 montre que la performance de DLC-M24 est satisfaisante par rapport à LC2-M24 et le MPC, malgré une plus grande valeur de la violation de la température. En effet, la température maximale à l'intérieur est proche de celle du cas MPC. La même remarque peut être notée pour la moyenne de température maximale quotidienne et pour la contrainte sur le mouvement du volet.

Consigne : $24\,^{\circ}\mathrm{C}$	Contrôleur optimal	LC2-M24	DLC-M24
Violation thermique [$^{\circ}$C.h/day]	1.47	1.79	2.04
Température maximale [$^{\circ}$C]	25.07	25.05	25.11
Moyenne de température maximale [$^{\circ}$C]	24.27	24.28	24.32
Violation visuelle [Lux.h/jour]	1930	1967	1967
Nombre de mouvement [par jour]	4.5	5.7	5.7
Cycle [par jour]	2.16	2.16	2.1

TABLE *4.9: Comparaison de la performance de contrôleurs simplifiés appris par différentes méthodes avec le contrôleur optimal avec la consigne à* $24\,^{\circ}\mathrm{C}$

Set-point : $26\,^{\circ}\mathrm{C}$	Optimal controller	LC2-M
Temperature violation [$^{\circ}$C.h/day]	4	5.06
Temperature maximum [$^{\circ}$C]	29.55	29.97
Mean Temperature maximum [$^{\circ}$C]	26.4	26.52
Luminance violation [Lux.h/day]	2755	2767
Blind movements [per day]	4.2	5.5
Cycle [per day]	2.18	2.34

TABLE *4.10: Résultats d'une longue simulation (d'avril au juin) avec la consigne à* $26\,^{\circ}\mathrm{C}$

4.3.5.4 Simulation pendant une longue période

Afin de valider cette approche pour la gestion des volets, nous considérons ici une longue simulation, i.e. d'avril à juin avec le contrôleur optimal MPC et le contrôleur simplifié LC2-M avec une consigne à $26\,^{\circ}\mathrm{C}$. Le tableau 4.10 montre la comparaison de la performance de ces contrôleurs. Nous voyons que les performances du MPC et du LC2-M sont encore très proches. Presque tous les critères sont dans les mêmes ordres de grandeur, bien qu'une différence de $0.5\,^{\circ}\mathrm{C}$ peut être remarquée pour le pic de la température intérieure ; c'est parce que LC2-M est appris à partir des données en mars, ce qui favorise la lumière naturelle. En tout cas, nous pouvons voir que le LC2-M fonctionne bien et rapproche ainsi les comportements du MPC avec un calcul minimal.

4.3.5.5 Portabilité de la solution

Dans cette section, nous cherchons à évaluer la portabilité de la solution en la testant dans d'autres conditions de simulation. En effet, LC2-M, qui a été appris précédemment, est mis en œuvre ici dans un autre bâtiment dont la surface vitrée est beaucoup plus petite, l'impact des flux solaire y est ainsi moins important. En outre, différentes conditions météorologiques (Trappes, France) sont également utilisées.

Nous montrons dans la figure 4.8 les résultats de simulation en juin où la température intérieure est très bien contrôlée à la consigne $26^{\circ}\mathrm{C}$. Étant donné que la surface vitrée est plus petite, l'effet de flux solaire sur la montée de la température est moins important. Ainsi, il convient de noter que le contrôleur approximatif décide de fermer complètement le volet très rarement. Pour les autres jours, il maximise la lumière naturelle en n'utilisant que la position fermeture à 50% et fermeture à 75%.

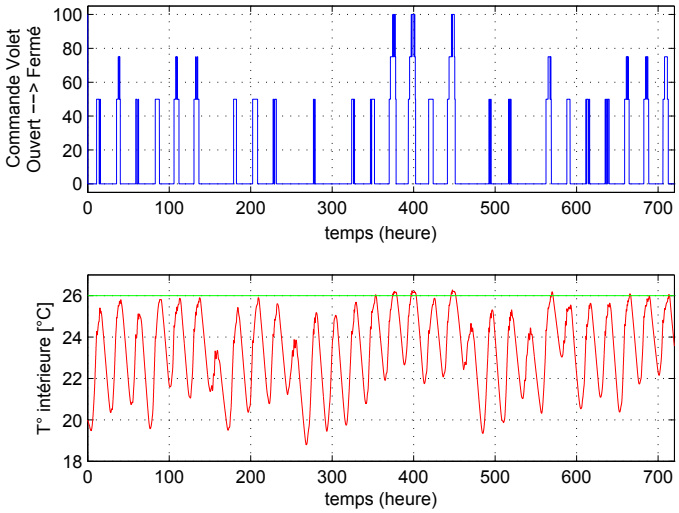

FIGURE *4.8: Résultats de simulation en juin avec des conditions météorologiques différentes et une maison différente. La consigne est mise à* $26°\mathrm{C}$

La méthodologie pour adapter la consigne en fonction des besoins des occupants est également testée dans les mêmes conditions, mais pour la consigne à $25°\mathrm{C}$. La figure 4.9 montre les résultats de simulations réalisées avec DLC-M25 dont les paramètres sont obtenus à partir des paramètres de LC2-M en diminuant le seuil de température β de $1°\mathrm{C}$.

La figure 4.9 montre que DLC-M25 décide de fermer plus souvent afin de respecter la consigne. La température intérieure est bien régulée et le compromis entre le critère de température et celui de lumière naturelle est aussi satisfaisant.

4.4 Conclusion

Dans ce chapitre, nous avons présenté une méthodologie pour construire un contrôleur très simple en apprenant les comportements du contrôleur optimal avec les outils du domaine Machine Learning tels que Arbre de décision, Adabost et SVM. Une comparaison entre différents outils a été présentée et des études plus poussées ont été aussi effectuées avec le SVM. Dans le cas de la gestion des occultations, il est démontré que les performances du contrôleur approximatif sont très satisfaisantes et très proches du contrôleur MPC, même avec un calcul plus simple. En effet, le contrôleur appris avec le SVM est construit avec seulement trois équations et deux règles logiques "si ... alors ... " . Il est ainsi très facilement mis en œuvre dans un microprocesseur pour être commercialisé.

Nous avons montré qu'une simulation pendant un ou deux mois de mi-saison semble suffisant pour constituer une base de données pour le problème d'apprentissage car les conditions météorologiques sont très variantes pendant cette période. De plus, nous avons souligné l'importance de bien choisir les mesures à prendre en compte en utilisant le "Feature selection". En effet,

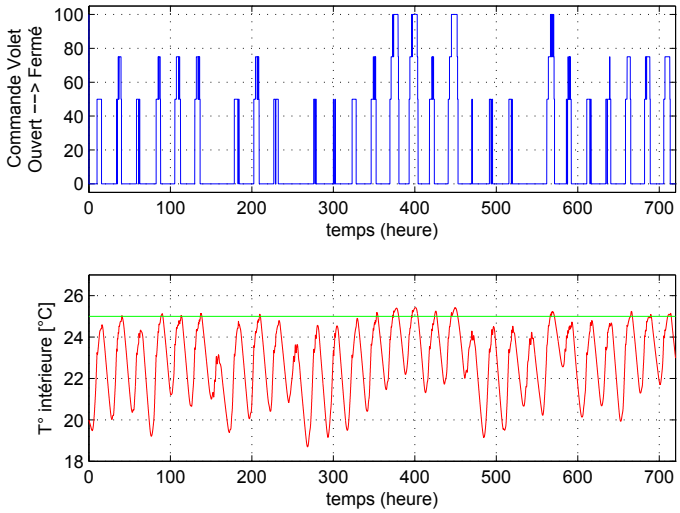

FIGURE *4.9: Résultats de simulation en juin avec des conditions météorologiques différentes et une maison différente. La consigne est mise à* $25°\mathrm{C}$*. Les paramètres du contrôleur pour cette consigne ont été adaptés à partir des paramètres du contrôleur de base pour la consigne à* $26°\mathrm{C}$

l'inclusion de mesures inutiles peut conduire à de moins bonnes performances.

Nous avons aussi proposé une nouvelle approche pour adapter les paramètres du contrôleur simplifié à de nouvelles consignes sans réapprendre tout le problème. Il est important de souligner que cette méthodologie n'est pas applicable avec d'autres techniques, en particulier Arbre de décision et AdaBoost. En outre, SVM semble mieux faire que ces dernières techniques en cas des données non séparables et surtout quand deux nuages de données sont mélangés. En effet, une ligne de séparation (deux paramètres) est suffisante pour classer avec 84% de précision pour les données de fermeture à 50% et 75% (cf. figure 4.7) tandis que AdaBoost a besoins de plus de 20 iterations, soit 40 paramètres, y compris les votes et les seuils.

Après une mise en œuvre plutôt satisfaisante de notre méthodologie concernant les systèmes d'occultations, nous visons à l'appliquer dans un cas d'étude plus complexe avec un contrôle coordonné des volets avec le système de ventilation qui fait l'objet de la prochaine partie de ce manuscrit.

Troisième partie

Cas d’application : Gestion des occultations et de la ventilation

CHAPITRE 5

Commande prédictive pour la gestion de la ventilation et des volets

Sommaire

5.1 Introduction **65**
5.1.1 Contexte 66
5.1.2 Objectif 67
5.2 Développement d'un contrôleur optimal MPC **67**
5.2.1 Modèles 67
5.2.1.1 Modèle thermique correspond au fonctionnement à vitesse 1 de la ventilation mécanique 69
5.2.1.2 Modèle thermique correspond au fonctionnement à vitesse 2 de la ventilation mécanique 69
5.2.1.3 Commutation de modèles 69
5.2.1.4 Modèle de ventilation naturelle 71
5.2.1.5 Modèle du CO2 73
5.2.1.6 Modèle de luminosité 75
5.2.2 Traduction mathématique des critères et des contraintes 75
5.2.2.1 Critères pour la gestion des occultants 76
5.2.2.2 Critères pour la gestion de la ventilation 76
5.2.3 Problème d'optimisation 77
5.2.4 Résolution du problème 79
5.3 Résultats de simulation **80**
5.3.1 Conditions de simulation 81
5.3.2 Période de mi-saison 81
5.3.2.1 Prévision parfaite 81
5.3.2.2 Prédictions imparfaites 83
5.3.2.3 Nombre d'occupants 84
5.3.3 Période d'été 85
5.4 Conclusion **86**

5.1 Introduction

Dans ce chapitre, notre méthodologie pour développer un contrôleur simple par un apprentissage à partir des résultats de simulation d'un controleur optimal sera étudiée pour un cas d'étude

plus complexe, celui du pilotage de la ventilation et des occultations. Les différents défis dans le développement d'un contrôleur optimal méritent d'être rappelés avant d'entrer dans le détail.

Trois phases de développement sont nécessaires pour construire un contrôleur MPC. Tout d'abord, il faut obtenir des modèles appropriés afin de bien prédire les évolution du comportement du bâtiment. Dans cette partie, le développement des modèles est bien plus compliqué que le cas d'étude précédent vu que nous avons trois systèmes installés dans la maison (les volets, la ventilation mécanique et la ventilation naturelle). Le défi suivant est comment définir les critères de confort pour l'occupant, à savoir dans ce cas d'étude, en plus des critères déjà présentés dans la partie précédente, il faut le critère de la qualité d'air et de la consommation. Ensuite, les pondérations de ces critère doivent être réglées, ce qui ne simplifie pas les choses car il faut trouver un compromis pour plusieurs critères. Enfin, est-ce que la méthode Branch and Bound est toujours convenable pour la résolution de notre problème d'optimisation car plus les variables de commandes sont nombreuses, plus la taille du problème est grande ?

La suite de ce chapitre va apporter des réponses pour toutes ces questions. Nous commençons par présenter le contexte de l'étude sur la qualité d'air qui devient problématique pour la santé des occupants.

5.1.1 Contexte

La qualité de l'air intérieur (QAI) est aujourd'hui reconnue comme un enjeu de santé publique tant au niveau international qu'en France [64, 65]. En effet nous passons en moyenne plus de 3/4 de notre temps dans des environnements clos et une majorité de ce temps dans l'habitat. De plus, les troubles de santé potentiellement associés à une mauvaise qualité de l'air intérieur sont nombreux et variés, et comprennent notamment les pathologies du système respiratoire (rhinites, bronchites) [66]. Une mauvaise qualité de l'air peut également favoriser l'émergence de symptômes tels que maux de tête, fatigue, irritation des yeux, nausées, etc. A l'inverse, une bonne qualité de l'air à l'intérieur d'un bâtiment a un effet positif démontré sur la diminution du taux d'absentéisme et peut favoriser le bien-être des occupants.

De nombreux facteurs jouent un rôle dans la dégradation de l'air intérieur du bâtiment tels que son emplacement, son ameublement et particulièrement la qualité de ce logement et son taux de renouvellement d'air [67]. En effet, la tendance actuelle de renforcer l'isolation du logement peut avoir un impact négatif sur la qualité de l'air intérieur. Les nouveaux bâtiments sont généralement plus isolants et plus étanches que les anciens. Le but de cette construction est de limiter les pertes via les parois mais elle réduit aussi l'échange de l'air de l'intérieur vers l'extérieur, causant ainsi des moisissures, un taux d'humidité important et une concentration de polluants, en particulier le CO_2 qui est un indice de la qualité de l'air. Pourtant les systèmes de ventilation sont souvent négligés dans la conception du bâtiment et ils sont souvent inadaptés et sous dimensionnés.

C'est pourquoi il est préférable d'être aidé par la ventilation naturelle, via par exemple une ouverture de fenêtre afin d'augmenter le taux de renouvellement d'air lors du pic de pollution dans l'habitat. Elle élimine ainsi les odeurs et diminue les concentrations en micro-organismes et polluants chimiques [68].

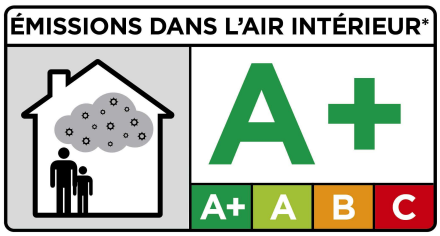

FIGURE *5.1: L'étiquette sanitaire des produits de construction et de décoration*

5.1.2 Objectif

L'objectif de ce chapitre est de présenter notre stratégie de commande optimale pour un fonctionnement efficace et coordonné de l'ensemble des systèmes des occultations et de la ventilation. Nos idées restent les mêmes en ce qui concerne le pilotage des occultations, i.e. utilisation de 4 positions de volet pour assurer le confort thermique, visuel et sonore de l'occupant. Nous avons aussi montré précédemment que pendant certaines journées, la température intérieure ne cesse d'augmenter malgré la protection contre l'énergie solaire via la fermeture des volets. C'est pourquoi, une bonne gestion coordonnée des volets avec la ventilation pourrait apporter une meilleure solution afin de garantir le confort thermique. Par exemple, la ventilation pourra fonctionner à une vitesse supérieure ou ouvrir la fenêtre pour déstocker la chaleur dans la pièce au lieu de solliciter les systèmes de climatisation.

De plus, nous abordons et traitons le problème lié à la qualité de l'air intérieur dont nous avons choisi le taux CO_2 comme indicateur. Par conséquent, nous aurons à notre disposition 3 modes de fonctionnement de la ventilation : la ventilation mécanique fonctionnant à deux vitesses et la ventilation naturelle.

5.2 Développement d'un contrôleur optimal MPC

La figure 5.2 présente le schéma de fonctionnement de notre contrôleur. Il est important de noter que contrairement au cas précédent où l'on a besoin uniquement d'un modèle thermique du bâtiment, il faut dans ce cas d'étude plusieurs modèles pour prédire l'évolution de la température intérieure et qui correspondent aux différents modes de fonctionnement de la ventilation. Un modèle de prédiction du niveau de luminosité ainsi que l'évolution du taux de CO2 dans la pièce sont aussi nécessaires pour assurer le confort visuel et la qualité de l'air intérieur. L'obtention des modèles sera présentée dans la section suivant. Ensuite, la formulation du problème d'optimisation sera détaillée et les résultats de simulation ainsi que des discussions seront présentés à la fin du chapitre.

5.2.1 Modèles

Dans cette section, nous présentons tous les modèles de prédiction nécessaires à la commande prédictive de la ventilation et du volet pour le confort de l'occupant.

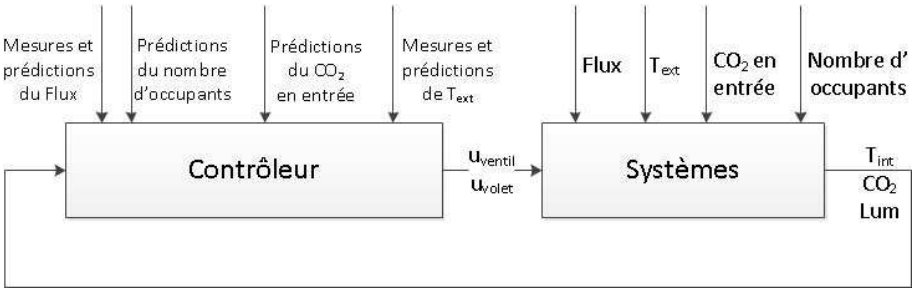

FIGURE *5.2: Schéma MPC pour la gestion des occultations et du système de ventilation*

Pour la commande de la ventilation, nous avons la possibilité de contrôler la ventilation mécanique qui fonctionne sous deux vitesses, une vitesse de base (vitesse 1) correspond à la réglementation et une deuxième vitesse avec un débit de renouvellement d'air plus important. De plus, nous pouvons aussi commander l'ouverture d'une fenêtre afin de créer une ventilation naturelle dans la maison. Pour le contrôle des occultations, les éléments nécessaires restent identiques à ceux détaillés précédemment.

Par conséquent, seulement 3 modèles sont nécessaires pour commander le volet et le système de ventilation. En parallèle, pour contrôler la qualité de l'air intérieur, trois modèles de l'évolution du CO_2 sont aussi nécessaires et correspondent à la capacité de renouveler d'air et ainsi faire changer le taux de CO_2 dans la pièce.

Nous commençons par détailler différents modèles thermiques de la pièce avec différents modes de fonctionnement de la ventilation. Ensuite, les modèles de prédiction du CO_2 seront présentés et la section se termine par les autres modèles nécessaires à la formulation du problème.

Nous avons présenté notre procédure d'identification du modèle thermique du bâtiment qui permet de prédire l'évolution de la température intérieure dans la pièce en fonction des conditions météorologiques (i.e. la température extérieure et l'énergie solaire reçue) ainsi que de la fermeture du volet (c.f Chapitre 3). Il est important de rappeler que le modèle développé dans le chapitre précédent correspond à un fonctionnement à petite vitesse (vitesse 1) de la ventilation.

La même démarche est appliquée afin d'obtenir les modèles thermiques correspondant aux autres modes de fonctionnement, i.e. ventilation à grande vitesse (vitesse 2) et ventilation naturelle. Pour cela, nous allons effectuer une simulation pendant une longue durée et ces données de simulation nous permettront d'obtenir des modèles de type "boîte noire" grâce au toolbox "identification" de Matlab.

Les modèles de la ventilation mécanique ainsi que leurs résultats de prédiction seront présentés en premier lieu. Le modèle de la ventilation naturelle nécessitera cependant une estimation du débit d'air qui correspond à une ouverture de fenêtre. Cet élément ainsi que les résultats montrant la qualité de prédiction seront présentés dans la dernière partie.

5.2.1.1 Modèle thermique correspond au fonctionnement à vitesse 1 de la ventilation mécanique

Il est important de rappeler que la ventilation mécanique fonctionne toujours et elle fonctionne à la vitesse 1 pendant notre période de simulation. Par conséquent, nous pouvons récupérer ce modèle déjà identifié pour notre problème de commande de la ventilation et des occultations. Pour plus de détails sur ce modèle, veuillez vous référer au chapitre précédent (section 3.2.1.1)

5.2.1.2 Modèle thermique correspond au fonctionnement à vitesse 2 de la ventilation mécanique

Nous sommes aussi partis du même principe pour obtenir un deuxième modèle discret correspond à un fonctionnement sous le régime sur-ventilé de la ventilation mécanique afin d'évaluer l'influence d'une augmentation de débit de renouvellement d'air sur la température intérieure.

Pour cela, nous utilisons les mêmes conditions de simulation que pour l'identification de la vitesse 1, c'est-à-dire une simulation pendant 2 mois avec les données météorologiques de Vaulx-en-Velen. Le volet reste ouvert et il n'y a pas d'occupant à l'intérieur de la maison.

5.2.1.3 Commutation de modèles

Contexte Nous avons choisi la méthode d'identification "boîte noire" pour obtenir nos modèles de prédiction de ventilation mécanique. Ces modèles sont bien adaptés pour être utilisés dans un problème de commande prédictive à condition que les états soient bien estimés. Dans nos cas d'étude, un reconstructeur d'état a été aussi implémenté afin de fournir une bonne estimation des états initiaux au modèle thermique. En revanche, il est important de souligner que cette méthode d'identification ne permet pas d'obtenir les matrices d'états d'origine (A,B,C) du système mais un autre tuple $(\hat{A},\hat{B},\hat{C})$ dont leurs relations peuvent être exprimées par :

$$\begin{cases} \hat{A} &= KAK^{-1} \\ \hat{B} &= KB \\ \hat{C} &= CK^{-1} \end{cases}$$

Pour la mise en œuvre de la commande prédictive pour les systèmes linéaires, cette non-unicité ne pose pas de problème étant donné que la reconstruction d'état ainsi que le modèle de prédiction peuvent être réalisés grâce à n'importe quelle représentation de matrices d'états. En revanche, dans ce contexte d'un système à commutation, chaque mode doit être exprimé selon le même vecteur d'état afin d'assurer une cohérence des prédictions.

En effet, considérons par exemple deux modèles correspondant à deux vitesse de fonctionnement de la ventilation, ils peuvent être présentés par :

— pour la vitesse 1 :

$$\begin{aligned} x_1(k+1) &= A_1 x_1(k) + B_1 u(k) \\ T_i(k) &= C_1 x_1(k) \end{aligned}$$

— pour la vitesse 2 :

$$\begin{aligned} x_2(k+1) &= A_2 x_2(k) + B_2 u(k) \\ T_i(k) &= C_2 x_2(k) \end{aligned}$$

Étant donné que nous n'avons aucune garantie de l'unicité de la base dans laquelle sont exprimés les états des modèles et que nous savons a priori qu'il y a continuité de la sortie et de l'état lors de la commutation entre différents modèles, il nécessaire de trouver un modèle général qui permet de satisfaire ces conditions. Ce modèle peut être représenté sous une forme suivante :

$$\begin{aligned} x(k+1) &= A_{\sigma(k)} x(k) + B_{\sigma(k)} u(k) \\ T_i(k) &= C x(k) \end{aligned}$$

Dans cette équation, $\sigma : \mathbb{R}^+ \to \Gamma$ est une loi de commutation qui indique le mode actif parmi un ensemble de mode de fonctionnement. La sortie du modèle est supposée être continue, par conséquent la matrice C ne dépend pas du mode actif.

Il est clair que si nous utilisons des modèles basés sur des états différents, ni les états ni la sortie ne sont continus et les modèles peuvent donc fournir de mauvaises prédictions. C'est pourquoi nous avons décidé de choisir un état identifiée comme l'état de référence. Nous voulons donc ramener les modèles des autres modes de fonctionnement dans cette même base. Pour ce faire, nous proposons par la suite une méthode simple et efficace pour déterminer la matrice de changement de base K_i qui permet d'exprimer tous les modèles de tous les modes de fonctionnement en une même base.

Méthode pour obtenir une matrice de changement de base Nous supposons que l'état du système est identique avant et après commutation et notre méthode est basée sur cette hypothèse. Pour obtenir la matrice de changement de base K, nous proposons de résoudre un ensemble d'équations issues des relations entre les états avant et après commutation. Il est important de noter que la taille de la matrice K dépend de l'ordre du système. Par exemple, si les systèmes à commuter sont d'ordre n, il faut donc trouver une matrice K de taille $n \times n$. Par conséquent, il faut au moins n relations pour rendre possible la résolution de l'équation. Même si un nombre n de couples d'états devrait suffire, il est préférable en pratique d'obtenir K avec un nombre de relations plus grand que n pour réduire la sensibilité aux changements d'états de ce dernier.

Le problème revient à estimer l'état initial de chaque mode de fonctionnement en fonction de base choisie grâce à un reconstructeur d'état. L'état estimé est ensuite comparé avec celui obtenu grâce au modèle. Nous détaillons par la suite notre méthodologie pour obtenir la matrice de changement de base K. La procédure est composée de trois étapes :

— Etape 1 - Simulation : Une simulation a été effectuée avec SIMBAD pendant 4 jours (180è à 183è) avec un changement de vitesse de ventilation toutes les 4 heures(cf. Figure 5.3). Sachant que nos modèles sont de l'ordre 4 et la taille de K est 4×4, cette période de simulation nous fournit un nombre largement suffisant de relations entre les états avant et après les moments de commutation.

— Etape 2 - Enregistrement des couples d'états : Au moment de changements de mode, les états des 2 modèles sont estimés et ils sont ensuite enregistrés (Figure 5.4). Des couples

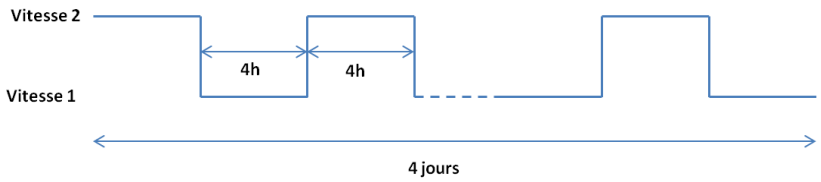

FIGURE *5.3: Créneaux de commande pour la génération de données*

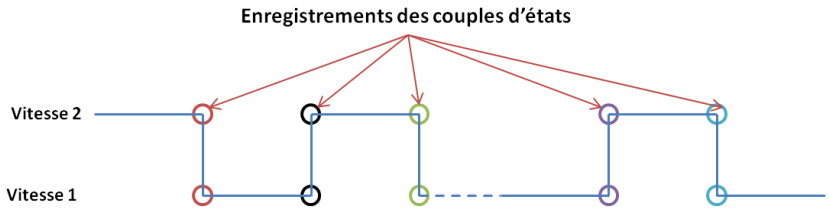

FIGURE *5.4: Enregistrement des couples d'états au moment du changement de consigne*

de données $(x_{(1,i)}, x_{(2,i)})$ sont obtenus, avec $x_{(1,i)}$ et $x_{(2,i)}$ sont respectivement l'état du modèle 1 et du modèle 2 au iè changement.

— Etape 3 - Détermination de la matrice de changement de base K : La matrice de changement de base K est enfin obtenue avec la méthode des moindres carrés.

Résultat de la mise en œuvre de la méthode Dans cette partie, nous présentons un exemple avec une prédiction pendant 4 heures (courbes rouges) avec deux changements de modes de fonctionnement pendant l'horizon de prédiction. La figure 5.5 présente une prédiction en après-midi, à partir de 19h. Des résultats satisfaisants ont été obtenus, sans sauts d'états, qui permet de valider notre méthode de recherche de la matrice de changement de base pour les systèmes à commutation. Il est important de souligner que ces résultats permettent aussi de vérifier à postériori la continuité de la sortie au moment de commutation.

5.2.1.4 Modèle de ventilation naturelle

La ventilation naturelle fonctionne grâce à deux phénomènes physiques qui sont la convection et le vent. En effet, l'air se déplace grâce aux différences de pression qui existent entre les façades du bâtiment et grâce à la différence de masse volumique en fonction de sa température.

La figure 5.6 montre la forme du courant d'air suite à une ouverture individuelle de fenêtre, ce qui permet de renouveler de l'air dans la maison.

Modélisation Il est important de noter que notre modèle du bâtiment deux zones dans SIMBAD n'est pas capable de simuler les phénomènes ainsi que les conséquence d'une ouverture de fenêtre.

Plusieurs études dans la littérature permettent d'estimer le changement de débit d'air via une ouverture de fenêtre. Pour notre cas d'étude, nous avons utilisé les résultats de [69] qui estiment le débit d'air dans le cas d'une ouverture individuelle de fenêtre par l'équation suivante :

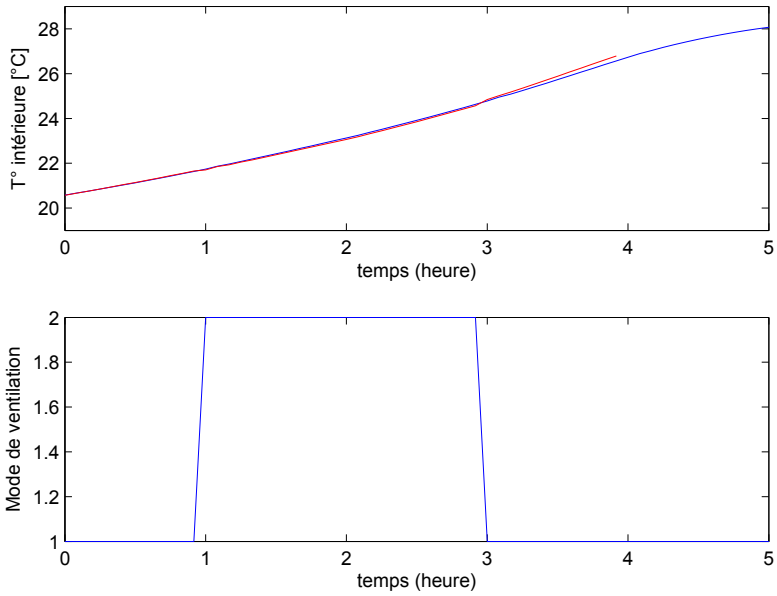

FIGURE *5.5: La prédiction avec le changement de base, comparée à la courbe réelles, 187e jour*

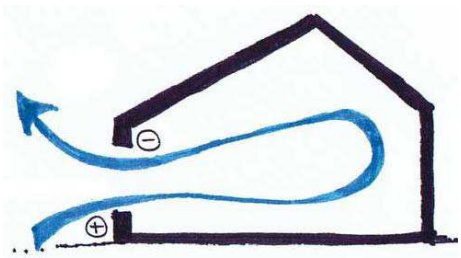

FIGURE *5.6: Exemple du courant d'air créé par une ouverture individuelle de fenêtre*

$$Q_v = A_{eff} \cdot U_m = \frac{1}{2} A \cdot U_m$$

avec :

— Q_v : le débit d'air $[m^3/s]$
— A_{eff} : la surface effective de l'ouverture $[m^2]$
— A : la surface totale de l'ouverture $[m^2]$
— U_m : la vitesse moyenne momentanée $[m/s]$ qui est estimée par :

$$U_m = \sqrt{(0.001 \cdot U_{10}^2 + 0.0035 \cdot h \cdot \Delta T + 0.01)}$$

avec :

- U_{10} : la vitesse du vent à 10 mètre de hauteur [m/s]

- h : la hauteur de l'ouverture [m]
- ΔT : la différence entre la température extérieure et intérieure

La prochaine étape est d'implémenter cette estimation de changement de débit d'air dans SIMBAD afin d'évaluer l'influence de la ventilation naturelle sur la température intérieure. Pour cela, nous effectuons une simulation pendant 30 jours avec les données suivantes :

— $A = 1.5\,m^2$
— $h = 0.5\,m$
— U_{10} : fourni par SIMBAD [m/s]
— ΔT : fourni par SIMBAD [°C]

Identification Le modèle de la ventilation naturelle peut ainsi être identifié comme pour la ventilation mécanique grâce à toolbox "identification" de Matlab. Il est important de noter qu'afin de pouvoir appliquer notre méthodologie de changement de base détaillée ci-dessus et ainsi d'éviter le soucis lié au changement de mode de ventilation, nous avons décidé de garder les mêmes vecteurs d'entrée et de ne pas prendre en compte dans notre modèle de commande d'autres facteurs d'influence comme par exemple la vitesse du vent, etc.

Un modèle discret de type "boîte noire" est ainsi obtenu :

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ T_{int}(k) &= Cx(k) \end{aligned}$$

avec $u(k)$ inchangé : $u(k) = \begin{bmatrix} \phi(k)(1-u_b(k)) \\ T_{ext}(k) \end{bmatrix}$

La figure 5.7 montre une bonne qualité de prédiction de 7.5 heures en avance de nos modèles en comparant avec une courbe réelle. Nous constatons aussi que la continuité des sorties est assurée même si nous basculons d'un modèle à un autre.

5.2.1.5 Modèle du CO2

Pour pouvoir prédire l'évolution du taux de CO2 dans les pièces, trois modèles doivent être développés qui correspondent à trois taux de renouvellement d'air. Nous avons à notre disposition des caractéristiques du système de ventilation grâce au fichier xml de SIMBAD. C'est pourquoi, nous avons décidé de construire directement un modèle numérique en se basant sur ces données, ce qui nous apporte l'avantage de bien contrôler les états des systèmes et des modèles.

La figure 5.8 montre le plan de la distribution de l'air de chaque pièce. En effet, une bouche d'air entrant est présente dans chaque pièce et l'air neuf, qui est représenté par des vecteurs verts, est distribué par part égale dans la chambre et le salon. Toute la partie de l'air dans la chambre est ensuite transférée vers le salon, ce qui est représenté par le vecteur orange. Enfin, l'air sort par la bouche d'air de sortie qui a été mise en place dans le salon.

Nous supposons que l'air qui rentre est égal à l'air qui sort de la maison, et en se basant sur l'équation de conservation de masse, l'évolution du taux CO_2 dans la pièce peut ainsi être estimée par :

$$\frac{\partial(VT_C)}{\partial t} = q_{in}C_{in} - q_{out}T_C + G$$

avec :

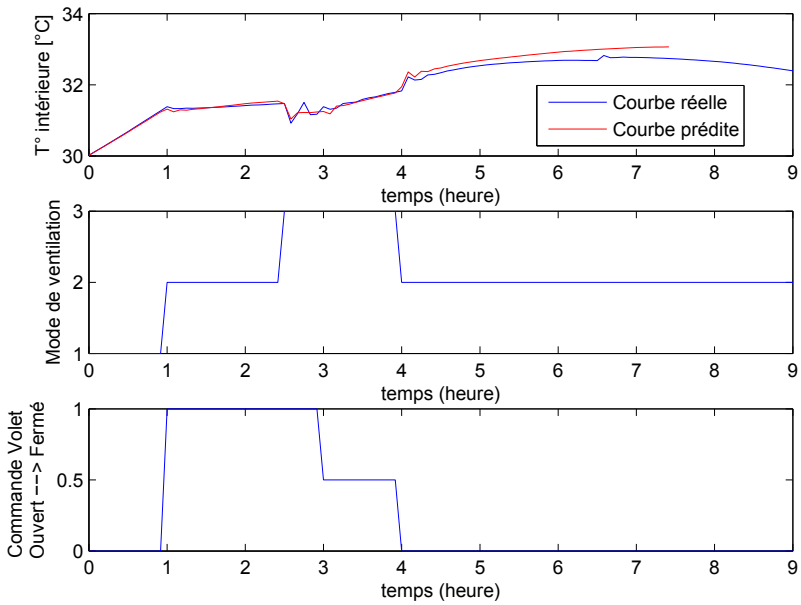

FIGURE *5.7: Comparaison d'une prédiction de température intérieure 7.5 heures en avance avec un fonctionnement mixte de différentes positions du volet et différents modes de ventilation*

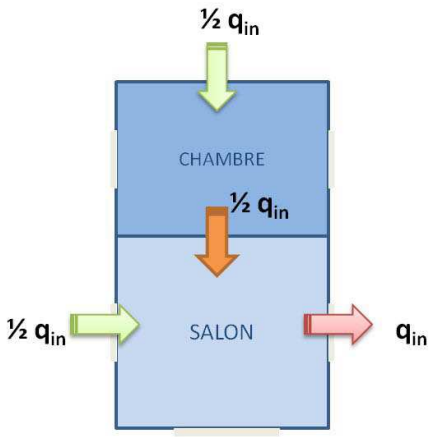

FIGURE *5.8: Plan de la distribution d'air de la maison*

— V : le volume de la pièce [m^3]
— T_C : le taux CO_2 dans la pièce [ppm]
— C_{in} : le taux CO_2 en entrée
— q_{in} : débit d'air entrant dans la pièce[kg/s]
— q_{out} : débit d'air sortant de la pièce [kg/s]
— G : génération de CO_2 de l'occupant [ppm]

Pour un volume constant de quantité d'air intérieure dans la pièce, l'équation précédente peut être écrite :

$$V\frac{dT_C}{dt} = q_{in}C_{in} - q_{out}T_C + G$$

En connaissant le débit d'air entrant et sortant, un modèle de type "boîte blanche" est ainsi obtenu. Dans notre cas d'étude, le débit d'air de notre système de ventilation est égale à $0.025\,kg/s$ en fonctionnement à petite vitesse et $0.035\,kg/s$ en fonctionnement à grande vitesse tandis que le débit de ventilation naturelle a été estimé à $0.1\,kg/s$.

Nous pouvons écrire sous la forme de modèle d'état comme ci-dessus :

$$\begin{aligned} \tilde{x}(k+1) &= \tilde{A}_i\tilde{x}(k) + \tilde{B}_i\tilde{u}(k) \\ T_C(k) &= \tilde{x}(k) \end{aligned}$$

avec :

— $\tilde{x}$: l'état du système à l'instant k

— T_C : le taux de CO_2 du salon

— $u(k) = \begin{bmatrix} n_{occ}(k) \\ C_{ext}(k) \\ C_{chambre}(k) \end{bmatrix}$ sont les entrées mesurables parmi lesquelles :

- $n_{occ}(k)$: le nombre d'occupants dans la pièce
- $C_{ext}(k)$: le taux de CO_2 à l'extérieure
- $C_{chambre}(k)$: le taux de CO_2 de la chambre à côté

5.2.1.6 Modèle de luminosité

Le modèle de luminosité reste identique au cas précédent dont l'expression est :

$$Lum(k) = G\phi(k)(1 - u_b(k))$$

avec G est le gain de conversion entre le flux solaire $[W/m^2]$ et la luminosité $[lux]$.

5.2.2 Traduction mathématique des critères et des contraintes

Dans cette section, notre formulation du problème d'optimisation est détaillée avec la partie de la gestion des occultations et aussi la partie pour la gestion de la ventilation. Il est clair que les deux systèmes contribuent tous à assurer le confort thermique de l'occupant. Le volet essaie de protéger le bâtiment contre une surchauffe due à un apport d'énergie solaire trop important tandis que la ventilation va sur-ventiler quand la température dépasse la zone de confort ou elle décharge de l'énergie stockée dans la pièce grâce à une ventilation naturelle pendant la nuit par exemple. Le but est toujours d'essayer de garder la température intérieure dans une zone de confort qui est définie entre $20\,°\mathrm{C}$ et $26\,°\mathrm{C}$.

Par conséquent, le critère sur le confort thermique commun pour les deux systèmes peut s'écrire comme ci-dessous :

$$J_{temp}(T_{int}) = \begin{cases} 20 - T_{int} & \text{if } T_{int} < 20 \\ 0 & \text{if } 20 \leq T_{int} \leq 26 \\ T_{int} - 26 & \text{if } T_{int} > 26 \end{cases}$$

avec T_{int} la température intérieure dans la pièce. Dans la partie suivante, nous présentons des différents critères qui sont liés à chaque système.

5.2.2.1 Critères pour la gestion des occultants

Nous rappelons dans cette partie deux autres critères principaux pour la gestion des occultants qui sont le critère du confort visuel et celui du confort sonore. Nous avons gardé les mêmes formulations que celles détaillées dans le chapitre 3.

Pour assurer le confort visuel, nous pénalisons le coût sur la luminosité lorsque son niveau devient inférieur à 300 lux :

$$J_{lum}(Lum) = \begin{cases} 0 & \text{if } Lum \geq 300 \\ 300 - Lum & \text{if } Lum < 300 \end{cases}$$

avec Lum est le niveau de luminosité dans la pièce

Et pour éviter des nuisances sonores liées à des mouvements intempestifs du volet, un changement de position du volet est pénalisé par le critère suivant :

$$J_{mov}(\Delta u_b(k)) = |u_b(k) - u_b(k-1)|$$

avec u_b est la position du volet

5.2.2.2 Critères pour la gestion de la ventilation

Après avoir rappelé les critères liés à la commande des volets, nous détaillons dans cette partie les critères propres à la gestion de la ventilation. En effet, l'autre avantage de la ventilation mécanique et naturelle est leur capacité de renouveler de l'air dans la maison, ce qui fait baisser ainsi le taux de CO_2 afin d'apporter une bonne qualité d'air intérieur pour l'occupant. Le seuil critique pour le taux de CO_2 est généralement fixé à 800 ppm. Par conséquent, nous avons décidé de pénaliser le dépassement du seuil critique. Le critère sur le CO_2 s'écrit ainsi :

$$J_{CO2}(T_C) = \begin{cases} 0 & \text{if } T_C \leq 800 \\ T_C - 800 & \text{if } T_C > 800 \end{cases}$$

avec T_C le taux CO_2 dans la pièce.

En parallèle, nous avons aussi un critère lié à la consommation d'électricité de chaque mode de ventilation. En effet, la vitesse 2 de la ventilation mécanique consomme plus d'énergie que quand elle fonctionne en vitesse 1 tandis que la ventilation naturelle via une ouverture de fenêtre

ne consomme quasiment rien. Par conséquent, le critère sur la consommation est obtenu grâce à l'équation suivante :

$$J_{conso}(v) = \begin{cases} 1 & \text{if } v = v_1 \\ 2 & \text{if } v = v_2 \\ 0 & \text{if } v = v_3 \end{cases}$$

avec v est le mode de fonctionnement de la ventilation :

— v_1 est le mode vitesse 1 de la ventilation mécanique
— v_2 est le mode vitesse 2 de la ventilation mécanique
— v_3 est le mode de ventilation naturelle.

5.2.3 Problème d'optimisation

Notre problème d'optimisation peut être formulé comme suit :

$$\begin{aligned} \min_{U_b(k:k+N_h-1),v} \sum_{j=1}^{N_h} \quad & [\, \gamma J_{mov}(\Delta u_b(k+j-1)) + \delta J_{lum}(Lum(k+j)) + \theta J_{temp}(T_{int}(k+j)) \\ & + \kappa J_{CO2}(C(k+j)) + \mu J_{conso}(v(k+j-1))] \end{aligned} \tag{5.1}$$

s.t : $\forall j = 1 \ldots N_h$,

$$\begin{aligned} x(k+j+1) &= A_{v(k+j)}x(k+j) + B_{v(k+j)} \begin{bmatrix} \phi(k+j)(1-u_b(k+j)) \\ T_{ext}(k+j) \end{bmatrix} \\ T_{int}(k+j) &= Cx(k+j) \\ \tilde{x}(k+j+1) &= \tilde{A}\tilde{x}(k+j) + \tilde{B} \begin{bmatrix} n_{occ}(k+j) \\ C_{ext}(k+j) \\ C_{chambre}(k+j) \end{bmatrix} \\ T_C(k+j) &= \tilde{C}\tilde{x}(k+j) \\ Lum(k+j) &= G\phi(k+j)(1-u_b(k+j)) \\ U_b(k+j-1) &\in \{0, 0.5, 0.75, 1\} \\ v &\in \{1, 2, 3\} \end{aligned} \tag{5.2}$$

avec $A_{v(k)}, B_{v(k)}, C$ sont les matrices d'états correspondant au mode de fonctionnement de la ventilation. $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ sont les matrices d'états du modèle de CO_2.

Les paramètres γ, δ, θ, κ, μ dans l'équation (5.1) sont des pondérations sur les différents critères et il est nécessaire de les bien choisir.

Pour le faire, nous supposons que le fonctionnement du système de ventilation (naturelle et mécanique) est indépendant de celui du système des occultations et ils n'agissent ensemble que sur le confort thermique. Par ailleurs, les volets agissent sur le confort visuel tandis que le système de ventilation intervient sur le confort CO_2. En effet, le fait de baisser le volet ne va pas influencer le taux de CO_2 dans la pièce et sur-ventiler ne va pas améliorer non plus le niveau de luminosité dans

la pièce. Grâce à cette hypothèse, nous décidons de régler la pondération de chaque système séparément en fixant la pondération du confort thermique et en variant la pondération d'autres critères de chaque système.

C'est pourquoi nous pouvons régler les pondérations du confort visuel δ et du confort sonore γ pour le système des occultations en restant sur le principe qui est déjà détaillé précédemment (cf. Section 3.2.3). En effet, en favorisant la luminosité dans la pièce, le volet a tendance à rester ouvert pour profiter de la lumière naturelle, ce qui fait aussi augmenter la température intérieure dans la pièce et qui provoque ainsi un inconfort thermique pour l'occupant. Nous avons montré une évolution contradictoire entre la courbe de violation thermique et celle de violation visuelle. Par conséquent, un compromis entre ces deux critères est obtenu avec la valeur correspondant à l'intersection de ces courbes.

Dans ce paragraphe, nous nous concentrons sur le détail du réglage de la pondération κ sur le confort CO_2 et μ sur la consommation pour le système de ventilation. Pour cela, nous décidons d'effectuer une simulation de 10 jours en mi-saison tout en faisant varier la pondération κ et en gardant celle du critère thermique θ constant. La figure 5.9 montre l'évolution de la violation normalisée du confort thermique et de la violation du confort CO_2 en fonction de différentes valeurs de κ.

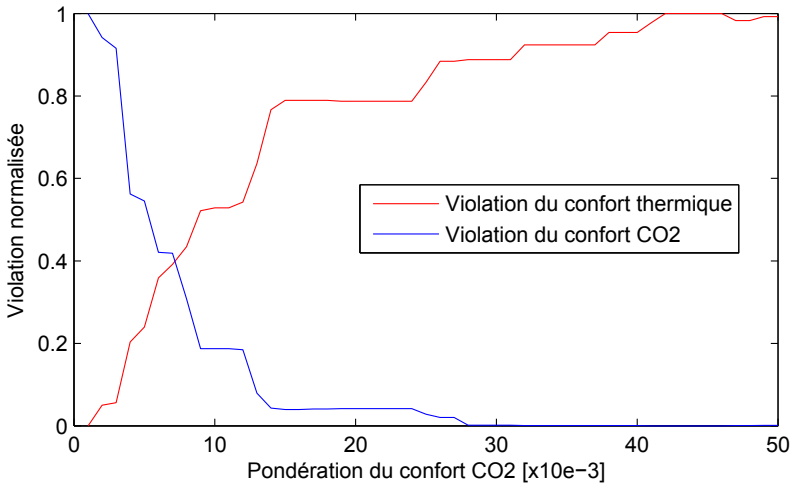

FIGURE *5.9: L'évolution de la violation du confort thermique et du confort CO_2 en fonction de la valeur de la pondération du confort CO_2*

Il est clair qu'en mettant le poids sur le confort sanitaire, le système de ventilation fonctionne souvent en mode ventilation naturelle, ce qui permet d'évacuer efficacement et de faire baisser rapidement le taux CO_2 dans la pièce. Pourtant, cela fait chuter aussi la température dans la pièce, surtout dans un climat mi-saison où la température extérieure est parfois très faible. Par conséquent, le confort thermique est gravement dégradé. C'est pourquoi, il est nécessaire de trouver un compromis entre le confort thermique et le confort sanitaire, ce qui est représenté par la valeur de κ correspondant à l'intersection de deux courbes.

De plus, il est aussi nécessaire d'optimiser la consommation d'énergie du système de ven-

tilation et de bien choisir la pondération pour la consommation. Pour cela, nous avons effectué une simulation avec différentes valeurs de la pondération sur le critère de la consommation puis nous regardons la consommation d'énergie sur ce période. L'ensemble de résultats obtenus est présenté par la figure 5.10.

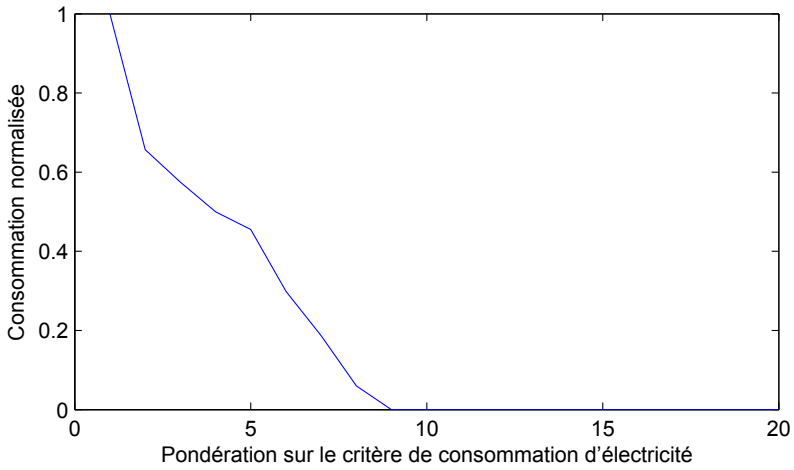

FIGURE *5.10: L'évolution de la consommation d'électricité normalisée en fonction de la pondération sur la consommation*

Il est clair qu'en mettant un grand poids sur le critère de consommation, le contrôleur décide de fonctionner principalement sous le mode de ventilation naturelle car ce dernier ne consomme pas d'énergie. Inversement, avec un poids faible de paramètre, la ventilation mécanique de vitesse 1 et de vitesse 2 est aussi sollicitée. Pourtant, une ouverture de fenêtre en permanence n'est pas non plus une solution optimale. En effet, nous n'avons pas pu prendre en compte dans notre problème d'optimisation des phénomènes qui sont très difficile à interpréter mathématiquement comme les bruits extérieurs, la pluie ou le vent violent. En effet, quand ces phénomènes apparaissent, nous avons tendance à fermer la fenêtre au lieu de ventiler naturellement. C'est pourquoi, un compromis se trouve sur la valeur de poids de 5, qui correspond à une consommation moyenne du système de ventilation.

5.2.4 Résolution du problème

Nous adoptons toujours la méthode de Branch and Bound pour résoudre notre problème d'optimisation. Il est important de rappeler que nous avons, pour ce cas d'étude, 4 positions différentes du volet avec 3 possibilités de fonctionnement de la ventilation. Par conséquent, il faut calculer au total les coûts d'une combinaison de 12 possibilités différentes à chaque instant.

Afin de réduire les efforts de calcul, nous avons décidé d'appliquer une technique qui consiste à chercher des fonctions constantes par morceaux de pallier T, avec $T = \alpha t_e$, $\alpha \in \mathbb{N}$, au lieu de chercher des fonctions par pallier t_e, qui correspond au pas d'échantillonnage du système.

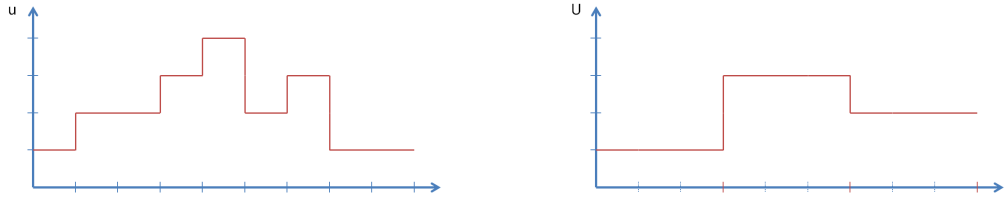

FIGURE *5.11: Augmentation du pas d'échantillonnage de commande pour calculer le critère afin réduire l'effort de calcul*

En effet, le problème d'optimisation

$$\min_{u(k:k+N_h-1)} \sum_{i=1}^{N_h} J(u(k+i-1))$$

va être écrit sous une nouvelle forme :

$$\min_{\mathrm{U}(k:k+N_h'-1)} \sum_{i=1}^{N_h} J(u(k+i-1))$$

s.l.c.[1] $\quad u(k+i-1) = \mathrm{U}(k+\lfloor \frac{i-1}{\alpha} \rfloor)$

avec[2] $\quad N_h' = \lceil \frac{N_h}{\alpha} \rceil$

La figure 5.11 montre un exemple avec $\alpha = 3, N_h = 9$ et le nombre de décisions est égale à 4. Nous voyons clairement l'intérêt de la méthode qui permet de réduire l'espace de recherche, dans notre cas d'étude, de $4^9(2.6*10^5)$ possibilités à $4^3(64)$ possibilités.

En parallèle, l'horizon de prédiction doit être bien choisi. En effet, nous avons montré qu'une prédiction d'environ 3.5 heures en avance est suffisante pour le pilotage des occultations. Nous avons décidé de prendre une valeur de 4 heures pour l'horizon de prédiction, qui garantie un temps de calcul raisonnable.

5.3 Résultats de simulation

Dans cette partie, nous présentons nos résultats de simulations obtenus avec le contrôleur optimal pour l'ensemble du système de ventilation avec le volet. D'abord, nos simulations sont effectuées avec des conditions météorologiques ainsi que le calendrier d'occupation parfaitement connus en avance. Ensuite, nous supposons que la présence des utilisateurs n'est pas connue, cette information est ainsi fournie uniquement par le capteur de présence et elle reste à la valeur mesurée sur tout l'horizon de prédiction.

Il est à rappeler que les résultats obtenus sont utiles par la suite dans le problème d'apprentissage. C'est pourquoi nous nous intéressons uniquement à deux périodes de mi-saison et d'été. Nous n'avons pas pris en compte la période d'hiver dans nos cas d'étude. En effet, les volets

1. $\lfloor \quad \rfloor$: partie entière
2. $\lceil \quad \rceil$: partie entière par excès

sont rarement sollicités pendant l'hiver vu que la température intérieure ne dépasse pas le seuil fixé à $26\,^{\circ}\mathrm{C}$. De plus, cela nous permet d'éviter les cas extrêmes où il n'y a pas d'autres choix que d'ouvrir la fenêtre pour faire baisser le CO_2 en même temps qu'activer le chauffage car il fait froid dehors. Il n'y a pas d'intérêt d'apprendre ces cas de figure. De plus, la taille du problème d'apprentissage peut ainsi être réduite.

5.3.1 Conditions de simulation

Les données météorologiques sont toujours les valeurs réelles mesurées à Vaulx-en-Velin en 2011. Dans ce cas d'étude, nous lançons l'optimiseur toutes les demies-heures pendant la simulation. Afin de réduire l'effort de calcul, nos prédictions sont sur 4 heures et les décisions sont effectuées toute les heures pendant la phase d'optimisation.

Nous étudions toujours la pièce au sud où la surface vitrée est importante. Pourtant, il est important de souligner qu'il y a un échange du CO_2 entre les deux pièces au nord et au sud. En effet, la pièce au nord est occupée le soir tandis qu'une présence de l'occupant dans la pièce au sud est souvent vue au début de la matinée, à la fin de la soirée ainsi que pendant le WE. La figure 5.12 montre les conditions extérieures en mi-saison, pendant 6 jours du mois de mars. Nous pouvons remarquer que le confort thermique est gravement dégradé sans la protection des volets. De plus, la qualité d'air n'est pas bonne non plus dans la pièce avec un pic de taux de CO_2 qui dépasse 1000 ppm. Il est important de noter qu'il y a 5 personnes dans la maison.

5.3.2 Période de mi-saison

5.3.2.1 Prévision parfaite

Dans cette partie, nos simulations sont effectuées avec une connaissance parfaite de l'évolution des conditions externes ainsi que de la présence de l'occupant dans chaque pièce. La figure 5.13 présente les résultats obtenus.

Quant au fonctionnement des occultations, la protection montre encore une fois son efficacité pour bloquer l'énergie solaire afin d'assurer un confort thermique de l'occupant.

Il est important de noter que la sur-ventilation mécanique n'est pas du tout sollicitée pour cette période. Il reflète aussi l'inefficacité de la ventilation mécanique simple flux à deux vitesses dans la réalité. En effet, la ventilation à grande vitesse consomme souvent le double par rapport à un fonctionnement à petite vitesse, mais pour un débit d'air qui atteint rarement le double. Ainsi, elle ne permet pas d'évacuer efficacement la chaleur ainsi que le CO_2 stockés dans la pièce.

Nous pouvons observer aussi que le contrôleur optimal décide de ventiler naturellement pendant la journée, il ouvre la fenêtre dès le matin où la température dépasse le seuil bas du critère thermique et il change la ventilation mécanique dans l'après-midi avant de baisser sous ce seuil. D'une part, c'est parce que le mode de ventilation naturelle est très favorisé car il ne consomme pas d'électricité. D'autres part, l'horizon de prédiction choisi pour notre contrôleur optimal est de quatre heures. Pourtant, c'est aussi à cause d'un court horizon de prédiction que le contrôleur a du mal à anticiper la baisse de la température intérieure dans la pièce pendant la nuit quand il fait froid dehors et cela entraine ainsi un inconfort thermique à la fin du troisième jour.

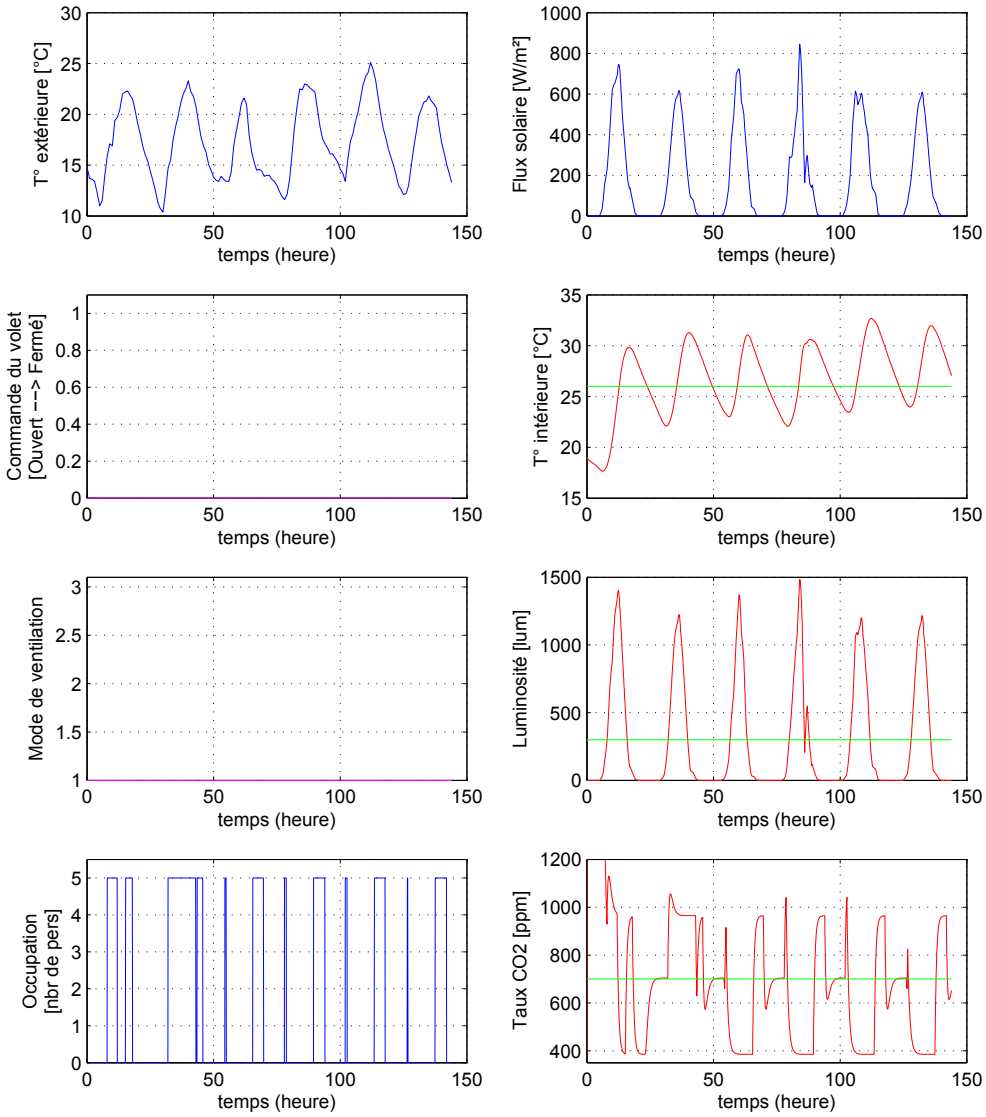

FIGURE *5.12: Conditions météorologiques et mesures de température intérieure et du CO_2 dans la pièce avec le volet restant ouvert et la ventilation fonctionnant à petite vitesse*

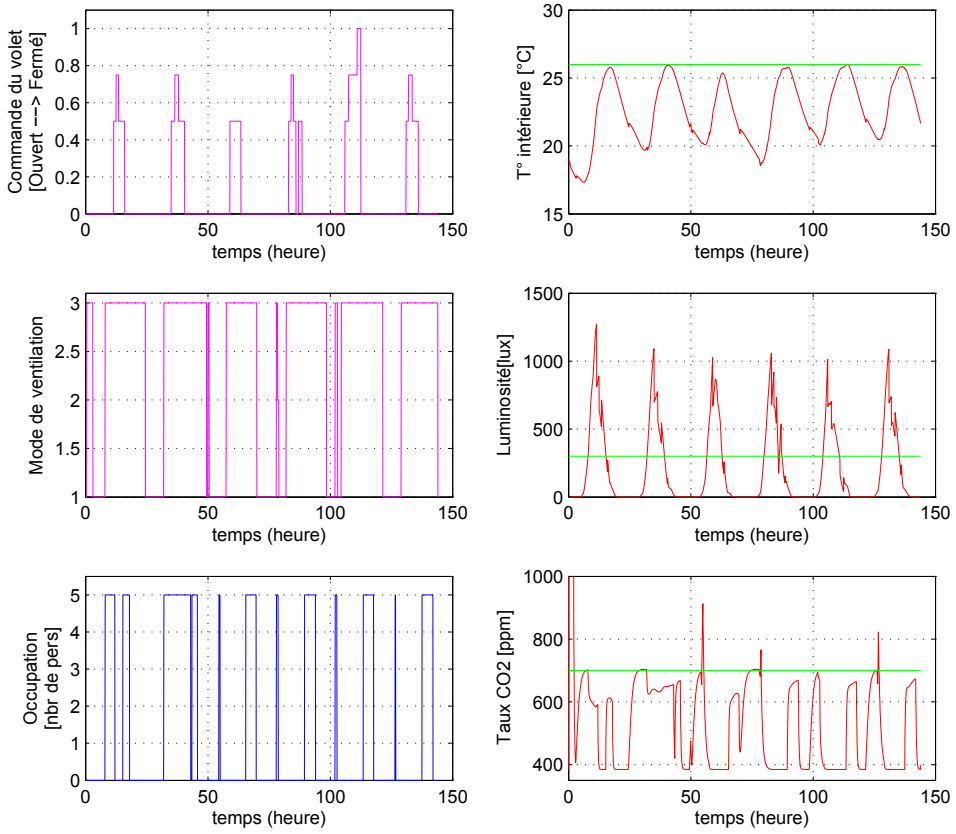

FIGURE *5.13: Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations pendant une période de mi-saison*

5.3.2.2 Prédictions imparfaites

Nous présentons dans cette partie les résultats de simulation obtenus quand les informations de la présence de l'occupant ne sont pas connues en avance. En effet, seul le capteur de présence fournit des données à l'optimiseur qui décide de garder ces dernières à la valeur mesurée sur tout l'horizon de prédiction.

La figure 5.14 montre les résultats pour cette même période de simulation avec les informations de présence manquantes.

On constate que contrairement à un cas parfait où le contrôleur est capable d'anticiper le retour de l'occupant dans la pièce, dans ce cas d'étude il a du mal à éviter l'inconfort lié à la qualité d'air à cause d'une montée rapide du taux de CO_2. Nous pouvons observer un taux de CO_2 qui se rapproche de 1000 ppm, surtout au début de la matinée où le taux CO_2 est déjà élevé dans le salon à cause du transfert de la chambre vers cette pièce.

Pourtant, il est important de noter aussi qu'au retour à la maison du travail de l'occupant en

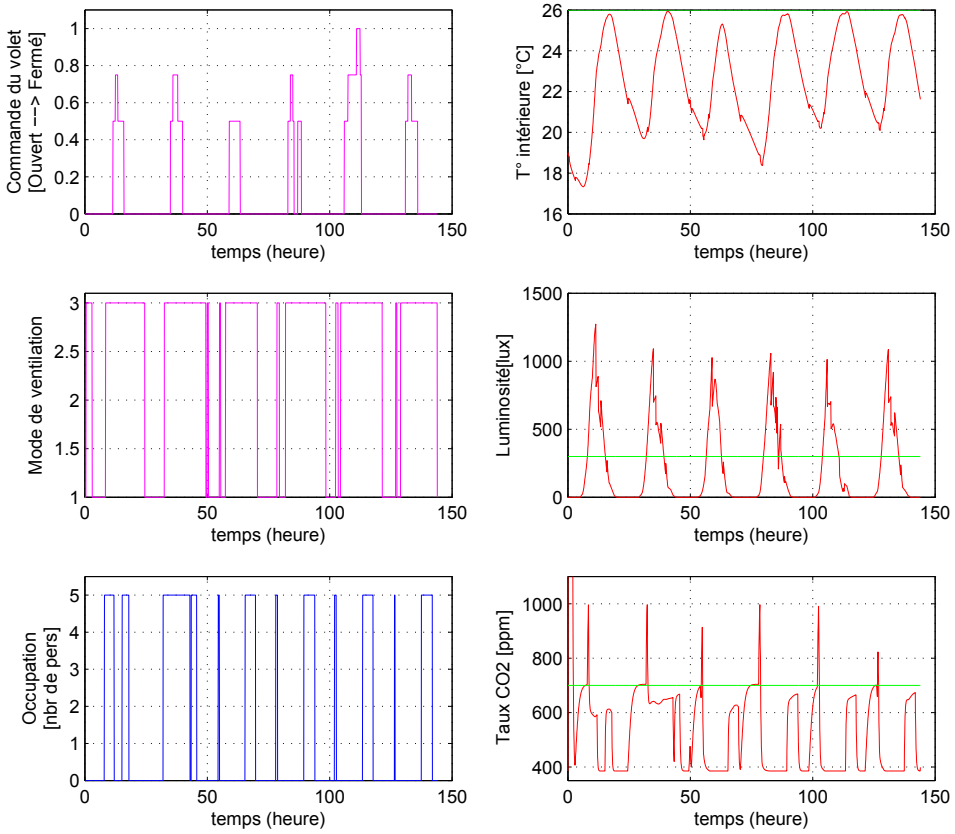

FIGURE *5.14: Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations*

fin de l'après-midi, le confort CO_2 est bien assuré car le taux qui est à 385 ppm est très faible au moment du retour. Par conséquent, l'ouverture de fenêtre peut compenser la production du CO_2 de 5 personnes dans la pièce.

5.3.2.3 Nombre d'occupants

Nous avons vu aussi que le nombre d'occupants est aussi un facteur important car l'évolution du taux CO_2 est proportionnel à ce nombre. Dans cette partie, nous présentons une simulation avec peu de personnes dans la pièce, i.e. deux personnes. Le résultat de simulation obtenu est montré par la figure 5.15.

Nous pouvons observer que dans ce cas d'étude, le contrôleur se concentre principalement sur le confort thermique avec une ouverture de fenêtre pendant la journée. Le taux de CO_2 est aussi bien maîtrisé et il est toujours en dessous du seuil à 700 ppm.

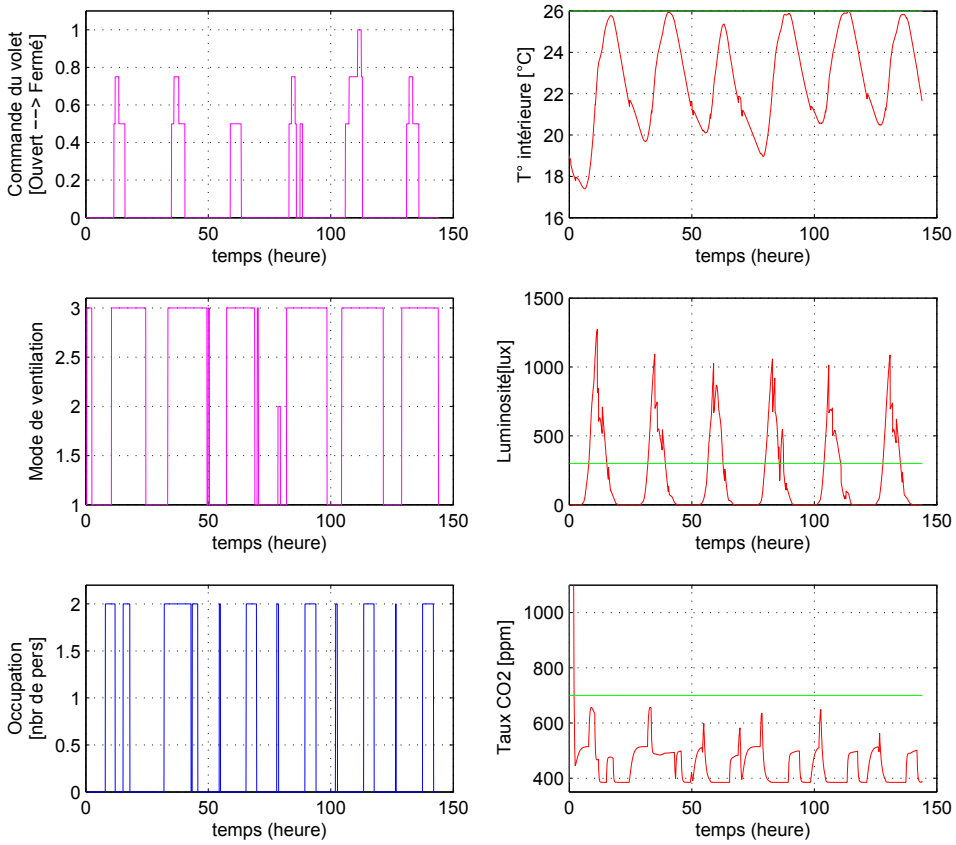

FIGURE *5.15: Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations dans le cas où il y a 2 occupants*

5.3.3 Période d'été

Maintenant nous nous intéressons à une période estivale afin de voir si notre contrôleur optimal se comporte différemment quand la température extérieure est élevée. La figure 5.16 montre les résultats de simulation obtenus.

Nous pouvons constater que la température intérieure des trois premiers jours est très bien régulée avec la protection des occultations contre l'énergie solaire. Pourtant, celle ci devient impuissante face à une température extérieure qui dépasse $30\,^{\circ}\mathrm{C}$ au quatrième jour, même avec une fermeture quasi totale à chaque instant.

Il est important de montrer que le contrôleur MPC décide de fermer la fenêtre quand il fait chaud dehors, afin de limiter le transfert de la chaleur de l'extérieur vers l'intérieur pour préserver la température intérieure dans la pièce. Ces moments peuvent être observés aux trois derniers jours de la période de simulation où la température intérieure dépasse le seuil souhaité qui est à $26\,^{\circ}\mathrm{C}$.

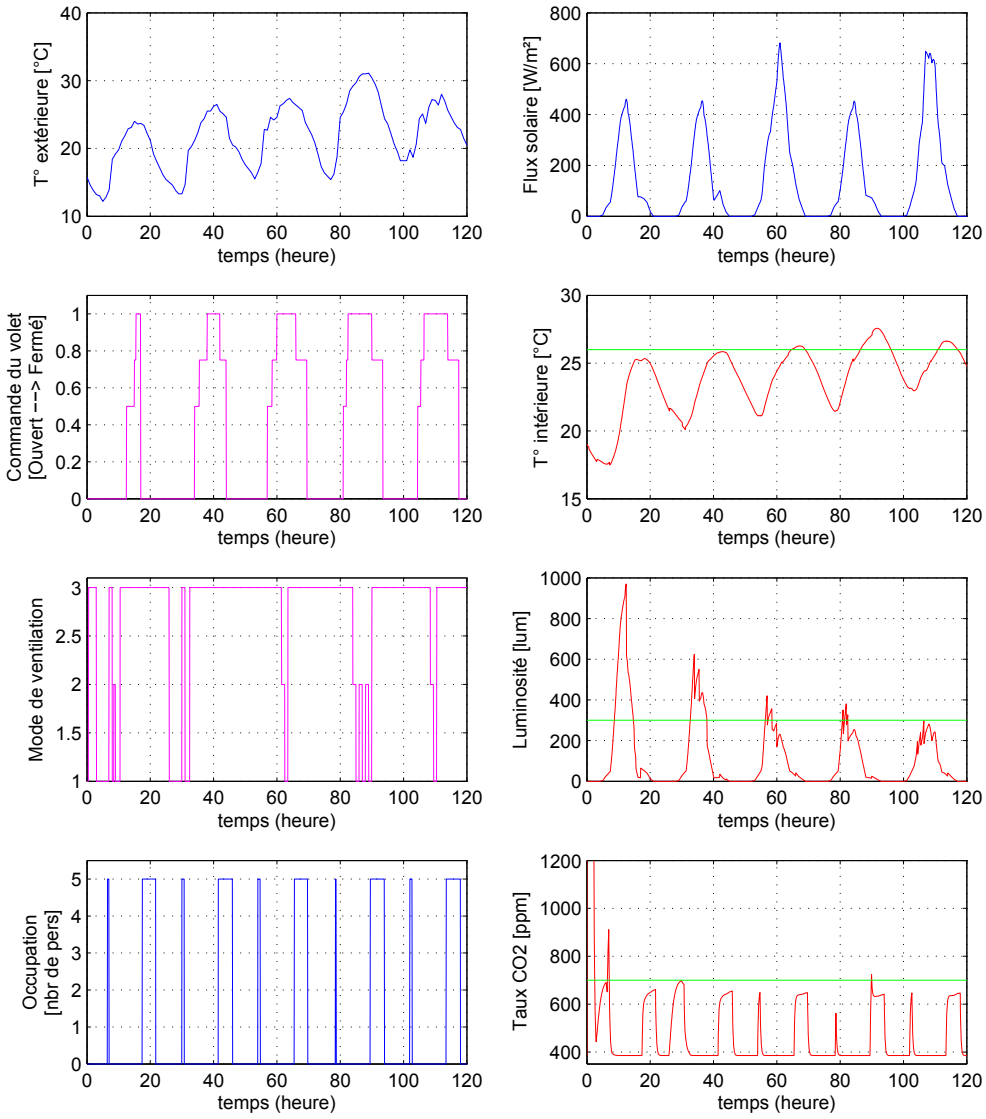

FIGURE *5.16: Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations pendant une période estivale*

5.4 Conclusion

Nous avons montré dans ce chapitre comment construire un contrôleur optimal MPC pour assurer différents conforts de l'occupant. Ceci est la première étape dans notre méthodologie du développement d'un contrôleur logique simple à base des règles logiques qui sont déduites d'un

apprentissage du comportement d'un contrôleur optimal.

Plusieurs défis ont été relevés. D'abord, nous avons réussi à réduire par 4 le nombre de modèles nécessaires pour la mise en œuvre de la commande MPC. De plus, la technique d'identification de type "boite noire" a été employée et elle a montré son efficacité pour remédier à la complexité d'obtenir un modèle de prédiction. Pourtant, les modèles obtenus ne sont pas exprimés dans une même base, ce qui pose des soucis en basculant d'un modèle à un autre pendant la phase de prédiction. Pour résoudre ce problème, nous avons proposé une méthode simple pour déterminer un changement de base.

La résolution du problème d'optimisation a été aussi étudiée sachant que la taille de ce dernier est beaucoup plus importante que celle du cas d'étude précédent. Pour cela, nous avons mis en œuvre une technique pour résoudre le problème d'optimisation et ainsi réduire l'effort de calcul nécessaire. De plus, le réglage des pondérations pour trouver un compromis pour notre problème multicritère n'était pas une tâche évidente. Nous avons proposé de régler les pondérations en fonction d'un critère commun de deux systèmes à commander, qui est le critère de confort thermique.

Nous avons proposé d'étudier dans ce chapitre le pilotage de façon optimale de l'ensemble de la ventilation et des occultations. A travers des résultats de simulation, nous avons aussi montré l'efficacité des occultations qui permettent de protéger la maison contre l'énergie solaire tout en limitant le nombre de changements de position ainsi que la durée d'une fermeture complète du volet, ce qui garantit un confort thermique, visuel et sonore pour l'occupant. De plus, le système de ventilation a cherché à assurer une qualité d'air satisfaisant en sollicitant la ventilation mécanique ainsi que la ventilation naturelle grâce à une ouverture de fenêtre. La ventilation contribue aussi à conserver une température agréable dans la maison en choisissant intelligemment les moments à ventiler afin de créer un courant d'air utile, quand la température extérieure est inférieure à la température intérieure.

Cette étude avec la commande prédictive nous donne ainsi quelques idées du fonctionnement optimal de l'ensemble des deux systèmes que nous avons essayé de comprendre et d'expliquer dans la partie qui présente les résultats de simulation. Le chapitre suivant nous éclaircira si nous pouvons retrouver des comportements de la commande prédictive avec un apprentissage automatique avec les algorithmes que nous avons choisis précédemment.

CHAPITRE 6

L'apprentissage du comportement du contrôleur optimal

Sommaire

6.1 Introduction **89**
6.2 Création d'une base de données **90**
6.3 Développement d'un contrôleur logique **91**
6.3.1 Sélection des attributs 91
6.3.1.1 Pour le système des occultations 92
6.3.1.2 Pour le système de ventilation 92
6.3.2 Apprentissage 93
6.3.2.1 Apprentissage pour les occultations 93
6.3.2.2 Apprentissage pour la ventilation 94
6.3.3 Résultats de simulation 95
6.3.3.1 Mi-saison 95
6.3.3.2 Eté 99
6.4 Conclusion **100**

6.1 Introduction

Ce chapitre correspond à la deuxième étape dans la méthodologie du développement du contrôleur simple à implémenter dans un produit industriel. Il présente donc comment on extrait des connaissances du comportement d'un contrôleur optimal afin de s'en servir pour construire le contrôleur logique dans le deuxième cas d'étude.

Notre but est de valider notre méthodologie avec l'étude d'un cas plus complexe, le fonctionnement de l'ensemble des systèmes de ventilation et des volets. Nous avons présenté dans la partie 2 que pour l'étude d'un simple fonctionnement du volet, le SVM est un bon outil pour apprendre et pour développer un contrôleur dédié à la protection de la maison contre les apports solaires. Nous cherchons à donner ici des réponses à différentes questions que pose le problème d'apprentissage : quel outil sera adapté pour ce cas d'étude ? Pourrons-nous ré-appliquer la méthode d'apprentissage avec le SVM pour le volet ? Quelle est la méthode pour apprendre le comportement de la ventilation ?

Différents éléments importants dans la chaîne de l'étude d'un problème d'apprentissage automatique seront présentés par la suite. Nous commençons par présenter la base de données

obtenue à partir d'une simulation avec le contrôleur optimal MPC. Ensuite, l'algorithme utilisé pour l'apprentissage du comportement optimal de nos systèmes sera détaillé. Enfin, différents scénarios de simulation seront dévoilés dans la dernière partie du chapitre.

6.2 Création d'une base de données

Notre base de données a été créée avec une simulation pendant 50 jours (de 150e jour au 200e jour) avec les conditions météorologiques présentées par la figure 6.1.

Nous avons montré que la base de données est un élément très important dans un problème d'apprentissage automatique. En effet, plus les données sont variées, plus l'apprentissage est efficace et il pourra prendre en compte plus de phénomènes.

Nous pouvons constater qu'avec ce choix de cette période de l'année, nous couvrons des jours de mi-saison quand la température extérieure varie entre $10\,°\mathrm{C}$ et $20\,°\mathrm{C}$. Nous y trouvons aussi des jours très chauds avec une température maximale qui peut atteindre $35\,°\mathrm{C}$. De plus, nous pouvons observer une grande variation du flux solaire dont la valeur maximale journalière se situe entre 100 W/m^2 pour un jour d'apports faibles, et 800 W/m^2 pour un jour sans nuage.

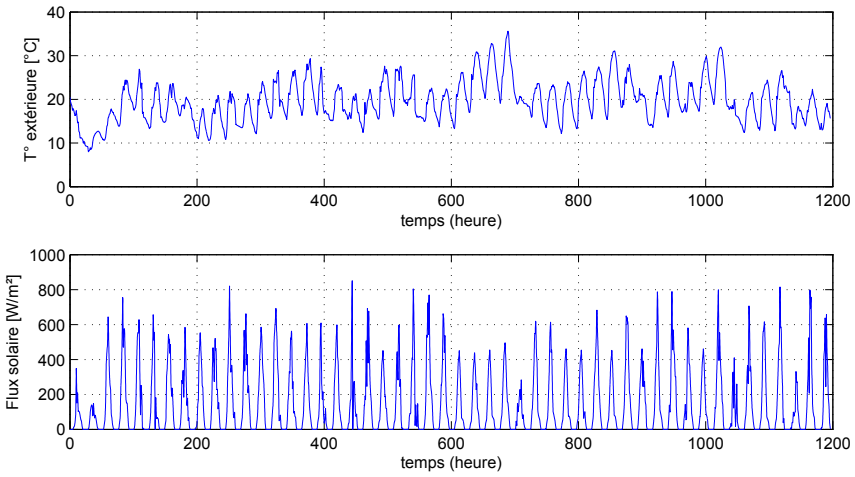

FIGURE *6.1: Principales données météorologiques de la période de simulation*

Les mesures venant des capteurs ainsi que les états du contrôleur, i.e. la position du volet et le mode de fonctionnement de la ventilation sont ensuite enregistrés toutes les 5 minutes afin de former une base de données. Les capteurs pris en compte dans le problème d'apprentissage sont :

- Capteur de température extérieure
- Capteur du flux solaire
- Capteur de température intérieure
- Capteur du taux de CO_2 intérieur
- Capteur de présence

— L'horloge

6.3 Développement d'un contrôleur logique

Dans cette partie, les algorithmes d'apprentissage automatique sont appliqués sur notre base de données afin d'extraire des connaissances utiles ou des règles logiques qui décrivent le comportement du contrôleur optimal.

La question est comment procéder à l'apprentissage de la synthèse du contrôleur logique. Devrions-nous effectuer un apprentissage global ? Nous pensons que non. D'une part, avec un fonctionnement complet du système de ventilation et du volet, la taille de la base de données devient grande. L'apprentissage d'un problème de grande taille n'est pas une tache aisée dans le domaine Machine Learning.

C'est pourquoi, nous avons supposé que le fonctionnement de la ventilation et celui des occultations sont indépendants, sauf pour assurer le confort thermique. Par conséquent, nous avons décidé d'étudier séparément deux systèmes, i.e. chaque système possède son propre problème d'apprentissage avec des attributs auxquelles ils sont attachés. Mais qui sont-ils ? Comment bien choisir ?

De plus, un apprentissage individuel des systèmes est plus adapté si nous voulons mettre en œuvre notre structure évolutive, qui permet d'intégrer ou enlever un système facilement de la structure, sans détériorer la performance et le fonctionnement des autres systèmes restants.

Afin de répondre à cette question, nous commençons cette partie par sélectionner les attributs utiles pour chaque problème d'apprentissage avec la "Feature selection". Ensuite, nous détaillerons notre méthode pour construire un contrôleur simple pour chaque système. Enfin, nous présenterons nos résultats de simulation afin d'évaluer la performance des contrôleur logiques et de comparer ces derniers avec le contrôleur optimal.

6.3.1 Sélection des attributs

Nous avons présenté trois algorithmes d'apprentissage automatique qui sont capables de choisir les meilleurs attributs à prendre en compte et de trier les attributs redondants, parmi lesquels figure la méthode SVM.

Nous avons décidé de faire le tri avec la méthode de SVM sur 6 attributs qui sont :

— La température intérieure
— Le flux solaire
— La température extérieure
— L'heure
— La présence de l'occupant
— Le taux de CO_2

Il est important de rappeler que la méthode SVM propose de classer l'importance des attributs selon le paramètre w de l'hyperplan. Plus la valeur de w est grande, plus l'attribut correspondant est important pour le problème d'apprentissage. Nous proposons par la suite d'analyser les attributs de chaque système.

	Flux	T_{ext}	T_{int}	Heure	Taux CO_2	Occupation
$H_{(1_075)}$	0.1	-9.4	-6.3	-2	-1.3	1.4
$H_{(075_05)}$	6.9	6.2	5.3	-0.7	-1.3	1
$H_{(0_05)}$	8.1	7.5	7.8	-1.5	-0.5	-0.4

TABLE *6.1: La valeur du paramètre w des hyperplans $H_{(1_075)}$, $H_{(075_05)}$, $H_{(05_0)}$*

	Flux	T_{ext}	T_{int}	Heure	Taux CO_2	Occupation
$H_{(v1_v3)}$	-0.15	-0.08	1.7	0.13	-5.7	3.2

TABLE *6.2: La valeur du paramètre w des hyperplans $H_{(v1_v3)}$ qui sépare la décision de la ventilation mécanique à petite vitesse et celle de la ventilation naturelle*

6.3.1.1 Pour le système des occultations

Nous appliquons ici la même méthode pour classer les attributs que celle déjà présentée pour le cas d'étude des volets seuls, i.e. nous regardons les paramètres w des différents hyperplans : $H_{(1_075)}$ qui sépare les décisions de fermeture complète à celles de fermeture à 75%, $H_{(075_05)}$ qui sépare les décisions de fermeture à 75% à celles de fermeture à 50% et $H_{(05_0)}$ qui sépare les décisions de fermeture à 50% à celles d'ouverture complète. Le tableau 6.1 présente les résultats correspondants.

Nous pouvons constater que l'heure, le taux de CO_2 et la présence de l'occupation ont peu d'importance sur la décision de fermeture ou d'ouverture du volet vu que leurs valeurs sont faibles par rapport au flux solaire, les températures intérieure et extérieure. En effet, choisir quelle mesure de capteur parmi ces trois derniers attributs est particulièrement difficile car leurs paramètres w des hyperplans sont souvent très proches.

Nous avons décidé de conserver notre choix initial de construire le contrôleur simple à partir de deux capteurs de flux et de température intérieure. Ce choix nous permet de comparer le contrôleur développé dans ce chapitre avec celui développé dans l'étude précédente où l'apprentissage a été fait à partir un pilotage optimal du volet tout seul.

6.3.1.2 Pour le système de ventilation

Le tableau 6.2 présente l'application du SVM afin de séparer à titre exemple la décision de fonctionner avec la ventilation mécanique à petite vitesse avec celle de fonctionner en mode ventilation naturelle.

Il est clair que le résultat obtenu avec le SVM nous suggère de considérer principalement les mesures suivantes :

— Le taux CO_2 dans la pièce
— La présence de l'occupant
— La température intérieure dans la maison

Ces attributs sont les plus importants pour le problème d'apprentissage. Il est intéressant de montrer que ces choix de mesures correspondent bien à nos constats faits précédemment dans la

Résultats SVM	Pente α	β
$L_{(05_0)}$	0.014	28.9
$L_{(075_05)}$	0.008	28.3
$L_{(1_075)}$	$6\ 10^{-4}$	25.6

TABLE *6.3: Valeur de la pente α et le paramètre β obtenus par un apprentissage avec le SVM*

partie simulation avec le contrôleur optimal. Nous avons remarqué que le contrôleur MPC décide d'ouvrir la fenêtre pour ventiler naturellement pendant la journée, au matin quand la température intérieure dépasse le seuil bas, i.e. $20\,^{\circ}\mathrm{C}$ et en fin de l'après-midi quand la température baisse, il décide de fermer la fenêtre avant d'atteindre ce seuil bas. De plus, la ventilation naturelle est aussi activée quand les occupants sont présents chez eux. En effet, afin d'anticiper le retour de l'occupant, il faut aussi ouvrir la fenêtre un peu en avant, d'où l'importance de la mesure du taux de CO_2 dans la pièce.

Quant au reste des mesures, leur poids est nettement moins important car elles ne sont impliquées que dans des cas rares, e.g. la fermeture de la fenêtre quand la température extérieure est importante même si la température intérieure dépasse le seuil de confort afin de bloquer le transfert de la chaleur chaude de l'extérieur vers l'intérieur etc.

6.3.2 Apprentissage

Dans cette partie, nous visons à développer des contrôleur logiques avec des connaissances du comportement du contrôleur optimal qui a été appris avec les algorithmes d'apprentissage automatique. Nous présentons par la suite la méthode utilisée pour le développement du volet ainsi que celui du système de ventilation.

6.3.2.1 Apprentissage pour les occultations

Après avoir effectué le tri pour choisir les meilleurs attributs pour le problème d'apprentissage pour les occultations, nous avons décidé de garder les mêmes attributs qui ont été utilisé dans le cas d'étude précédent : le flux solaire et la température intérieure. En effet, le fait de rester toujours avec SVM nous permet de comparer le contrôleur déduit d'un fonctionnement optimal d'un ensemble de différents systèmes avec celui déjà obtenu précédemment, i.e. étude du seul système des occultations.

C'est pourquoi, nous gardons la structure sous forme de machine d'état afin de gérer le changement de position du volet, qui est décidé grâce à différentes droites $L_{(1_075)}$, $L_{(075_05)}$, $L_{(05_0)}$ obtenues avec le SVM. Par conséquent, nous caractérisons la droite de séparation par sa pente α et une température de seuil β. Le tableau 6.3 montre les valeurs des paramètres α et β obtenus avec le SVM. La comparaison de ces paramètres obtenus par les deux cas d'étude est faite plus loin.

6.3.2.2 Apprentissage pour la ventilation

Nous présentons dans cette partie le résultat d'apprentissage pour le système de ventilation que nous avons obtenu avec la méthode de l'arbre de décision. En effet, il est clair que la méthode SVM linéaire favorise l'interprétation des résultats de l'apprentissage, en particulier dans le cas simple de la gestion des occultations. En revanche, nous avons constaté que les données liées aux décisions du système de ventilation ne sont pas bien séparés avec un simple hyperplan. De plus, nous avons plusieurs attributs qui interviennent dans ce problème d'apprentissage. C'est pourquoi, nous devons faire face à un problème de grande dimension qui complique la convergence pour trouver la solution.

La solution de l'arbre de décision a été choisie pour ce cas d'étude de par sa facilité d'interprétation du résultat d'apprentissage et la simplicité de mise en œuvre de la solution. Il convient de souligner aussi qu'il n'existe pas un algorithme magique qui pourrait convenir à tous les problèmes. En effet, une première analyse par l'humain avec sa bonne connaissance des données à apprendre est cruciale pour choisir l'outil et la méthode d'apprentissage.

La figure 6.2 montre la structure de l'arbre obtenu avec Weka en appliquant l'algorithme C4.5.

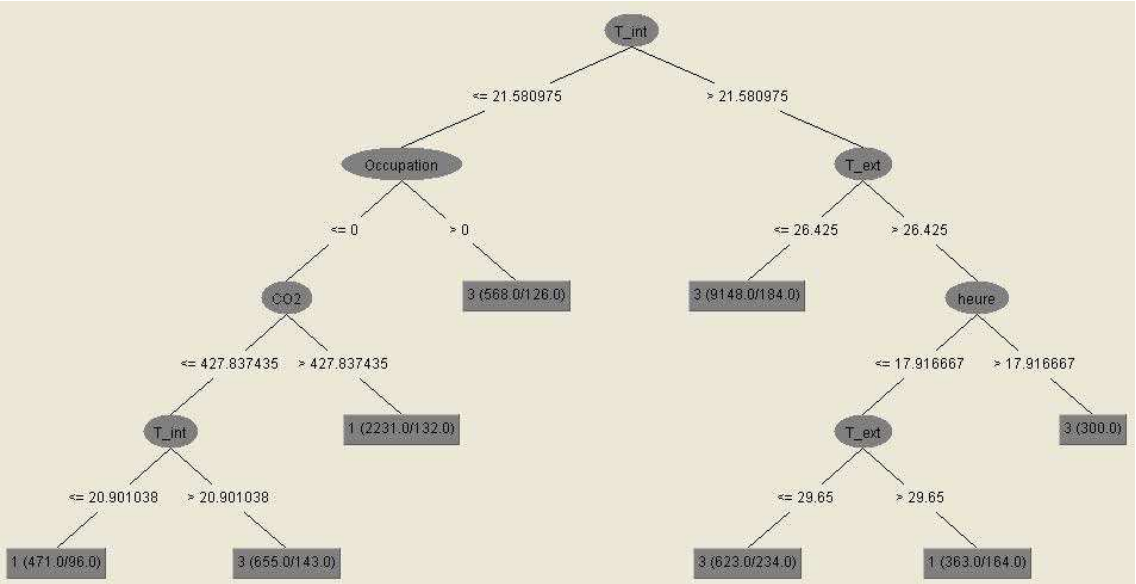

FIGURE *6.2: L'arbre de décision obtenu par l'apprentissage d'un contrôleur optimal pour le système de ventilation*

Nous pouvons remarquer deux saisons distinctes dans la structure de l'arbre. En effet, la branche droite représente la saison "Eté" où la température intérieure dépasse souvent $22\,^{\circ}\mathrm{C}$. Nous pouvons y observer la règle qui décide de limiter le transfert de chaleur chaude qui vient de l'extérieure en fermant la fenêtre quand la température extérieure dépasse $29.5\,^{\circ}\mathrm{C}$. Par ailleurs, nous pouvons remarquer aussi que nous avons sélectionné presque toutes les mesures d'entrées dans l'arbre de décision, sauf le flux solaire, tandis que pour la sélection faite avec le SVM nous avons indiqué que la mesure de la température extérieure ou l'heure ne sont pas importants. A notre avis, la raison est que le choix d'hyperplan linéaire pour effectuer la sélection avec SVM

n'était pas adapté dans ce cas où les données sont difficilement séparables.

Inversement, la branche gauche de l'arbre représente la saison plutôt mi-saison ou quand c'est la nuit car la température intérieure est faible. Nous y trouvons des règles qui s'intéressent plutôt à faire face au problème de la qualité d'air. En effet, il est décidé d'ouvrir la fenêtre quand le taux de CO_2 est inférieur à 430 ppm même si l'occupant n'est pas à la maison. Cette action traduit notre constat de l'anticipation du contrôleur optimal avant le retour de l'occupant. Par ailleurs, la ventilation naturelle fonctionne quand les utilisateurs sont chez eux.

Il est important de montrer aussi que la décision de fonctionner avec la grande vitesse de la ventilation mécanique n'apparait pas dans l'arbre de décision parce que le nombre de données de ce fonctionnement est beaucoup plus faible que le mode ventilation mécanique à petite vitesse ou celui de ventilation naturelle.

Nous avons décidé d'implémenter cette structure de l'arbre afin de construire notre contrôleur logique. La partie suivante présentera les résultats de simulation obtenus avec nos deux contrôleurs, i.e. le contrôleur SVM pour les occultations et le contrôleur de l'arbre de décision pour le système de ventilation.

6.3.3 Résultats de simulation

Dans cette partie, les résultats seront présentés pour deux saisons, i.e. la mi-saison et l'été. Différents scénarios de simulation seront présentés. Nous commençons par montrer les résultats obtenus en mi-saison avec un fonctionnement de l'ensemble des occultations et le système de ventilation. Ensuite, nous répondrons à la question de savoir s'il y a des conséquences quand l'utilisateur décide d'activer seulement le contrôle automatique des occultations en laissant la ventilation fonctionner à la vitesse 1. Nous voulons aussi comparer les résultats donnés par le SVM pour le volet dans cette étude avec ceux obtenus dans la partie précédente par l'apprentissage ce système tout seul, sans la ventilation. Enfin, une simulation en été sera présentée afin de vérifier si les résultats obtenus avec le contrôleur logique sont proches de ceux obtenus avec le contrôleur optimal quand il fait chaud dehors.

6.3.3.1 Mi-saison

Nous présentons ici les simulations pendant une période de mi-saison où la température extérieure se trouve principalement entre $15\,^{\circ}\mathrm{C}$ et $25\,^{\circ}\mathrm{C}$ afin de voir si les contrôleurs logiques sont capables d'assurer les différents conforts de l'occupant.

Fonctionnement de l'ensemble des occultations et du système de ventilation La figure 6.3 montre une simulation pendant 10 jours, à partir du 240e jour de l'année, avec le contrôleur logique pour les occultations et le système de ventilation.

Quant au fonctionnement du système de ventilation, nous pouvons observer que la qualité d'air est bien assurée dans la pièce, avec quelques moments où le taux de CO_2 dépasse un peu le seuil fixé à 700 ppm. Nous pouvons remarquer aussi que la fenêtre est fermée souvent quand la température intérieure risque de descendre en dessous du seuil à $20\,^{\circ}\mathrm{C}$.

Quant au contrôleur logique appris pour les occultations, il est montré que la décision basée sur deux mesures de capteurs (capteur de flux solaire et capteur de température intérieure) est

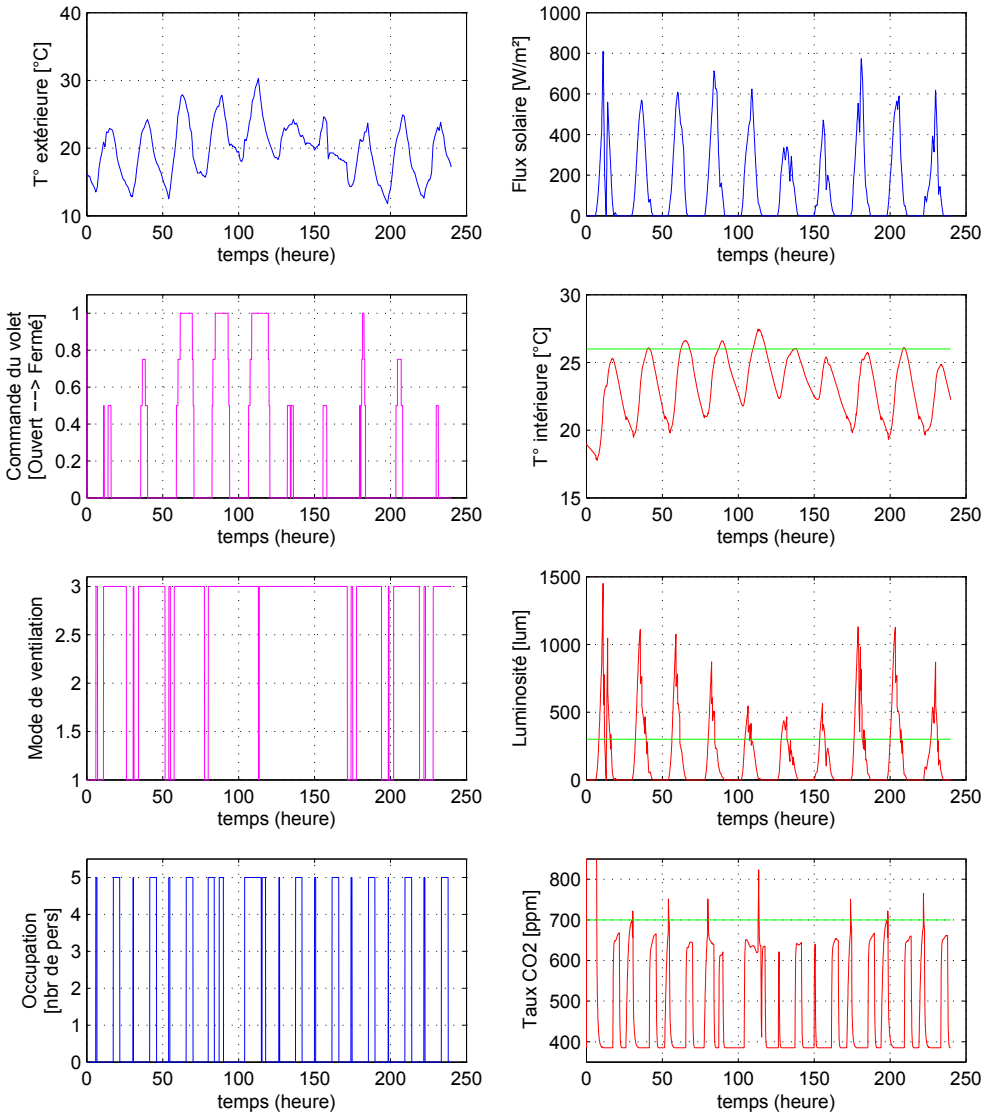

FIGURE *6.3: Résultats de simulation obtenus avec le fonctionnement de l'ensemble des occultations et le système de ventilation*

toujours efficace pour nous protéger contre les apports solaires. Nous pouvons remarquer que la température intérieure dépasse le seuil de confort à $26°\mathrm{C}$ au 5e jour malgré une fermeture complète du volet.

Résultats SVM	Cas d'étude simple		Cas d'étude complexe	
	Pente α	β	Pente α	β
$L_{(05_0)}$	0.0089	27.6	0.014	28.9
$L_{(075_05)}$	0.0014	25.9	0.008	28.3
$L_{(1_075)}$	$3.69\ 10^{-4}$	26.1	$6\ 10^{-4}$	25.6

TABLE *6.4: Comparaison de la pente α et le paramètre β des résultats obtenus par un apprentissage dans les deux cas d'étude*

Comparaison des deux contrôleurs simples obtenus dans les deux cas d'étude Dans cette partie, nous voulons comparer les deux contrôleurs que nous avons obtenus par l'apprentissage dans les deux cas d'étude, i.e. le cas simple où le volet a été étudié tout seul et le cas complexe où il y a aussi des stratégies de contrôle du système de ventilation.

Le tableau 6.4 compare les paramètres, i.e. la pente α et la température de seuil β des deux contrôleurs. Nous montrons les valeurs des paramètres obtenues par l'apprentissage en juillet pour le cas d'étude simple et celles obtenues pour le cas d'étude complexe avec une simulation pendant une période similaire dans l'année.

Nous constatons que les pentes des droite $L_{(05_0)}$, $L_{(075_05)}$ et $L_{(1_075)}$ appris pour le cas d'étude complexe sont toutes plus grandes que celles apprises pour le cas d'étude simple, ce qui entraîne ainsi une température de seuil plus grande aussi. La raison est que la température intérieure enregistrée pour le cas complexe est plus faible à l'aide du système de ventilation, surtout au début de la matinée. C'est pourquoi, il nécessite plus d'énergie solaire pour chauffer le bâtiment, ce qui implique que le contrôleur logique décide de fermer plus tard le volet.

La figure 6.4 montre les résultats de simulation obtenus avec le contrôleur logique déduit de l'étude précédent. Il est à noter que ce contrôleur a été appris en juillet, qui correspond à la période utilisée pour l'apprentissage du contrôleur complexe.

Nous pouvons voir que la courbe de température obtenue ressemble à celle montrée par la figure 6.3. Il est à noter quand même que la température intérieure dépasse un peu plus le seuil de confort car le volet est fermé plus tard à cause de la valeur de seuil de température plus petit ($26.1\,^{\circ}\mathrm{C}$ au lieu de $25.6\,^{\circ}\mathrm{C}$ pour le cas d'étude complexe). De plus, le fait d'avoir dans le cas d'étude complexe le système de ventilation ainsi que plus de critère à satisfaire que dans le cas simple, a une certaine influence sur les résultats d'apprentissage et pose ainsi plus de problème à l'apprentissage.

Fonctionnement du volet tout seul, sans le système de ventilation La figure 6.5 montre une simulation qui a été effectuée dans les mêmes conditions que l'étude précédente, mais avec le fonctionnement unique du volet pour éviter la surchauffe. La ventilation mécanique fonctionne en mode petite vitesse. Nous n'avons pas sollicité l'aide de la ventilation naturelle, ni de la ventilation mécanique à grande vitesse.

Nous pouvons observer que la température intérieure obtenue avec cette simulation est très similaire à celle obtenue dans la partie précédente (cf. Figure 6.3). Pourtant, le volet se ferme beaucoup plus ici, ce qui entraine un confort visuel moins confortable que dans le cas précédent.

Il est clair que la qualité d'air est sérieusement dégradée sans la ventilation naturelle, le taux

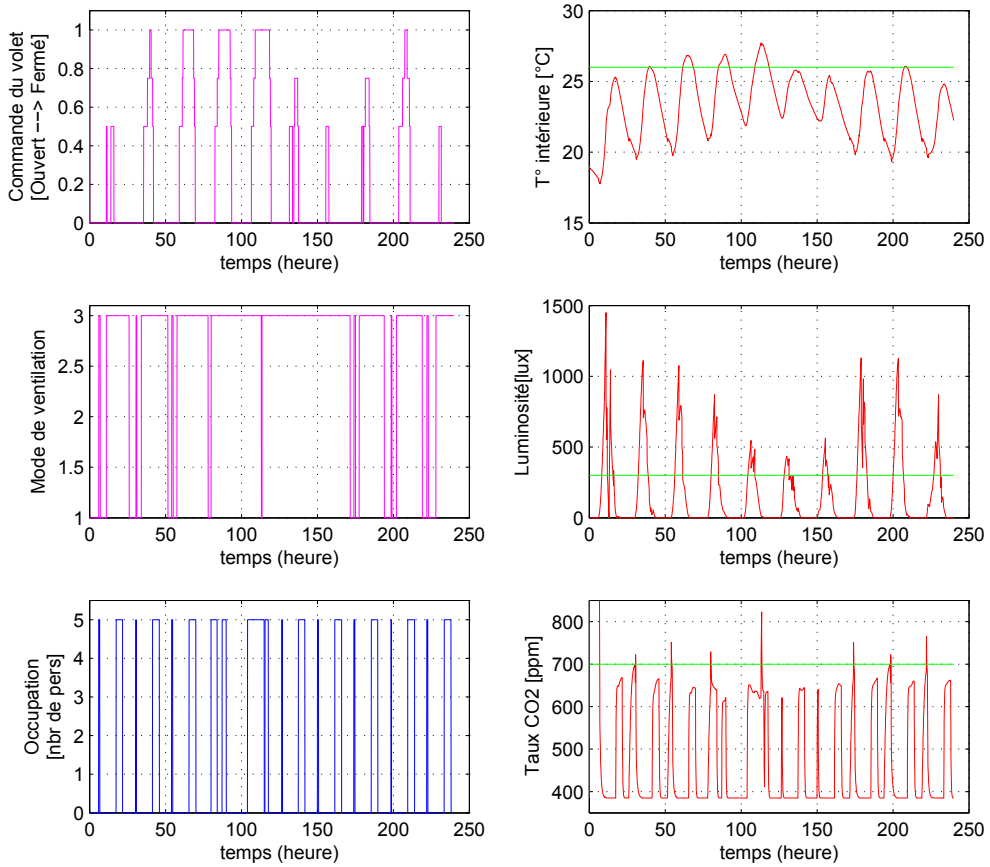

FIGURE *6.4: Résultats de simulation obtenus avec le fonctionnement de l'ensemble des occultations et le système de ventilation. Les paramètres du contrôleur logique pour le pilotage des volets viennent de l'étude précédente*

de CO_2 peut atteindre 1000 ppm. Il est préférable de rappeler qu'il y a 5 personnes dans la pièce. Ce résultat montre l'importance d'ouvrir la fenêtre quand nous sommes nombreux dans la pièce.

L'efficacité de la ventilation naturelle a été montrée avec ces résultats de simulation. En effet, non seulement elle permet d'assurer une bonne qualité d'air intérieur, mais elle vient aider aussi les occultations en ouvrant la fenêtre afin de faire baisser la température intérieure. Le volet reste ainsi plus souvent ouvert, ce qui permet de profiter de plus lumière naturelle.

Il est important de souligner aussi que la performance obtenue avec un fonctionnement indépendant des occultations et du système de ventilation est acceptable. La robustesse et l'efficacité de notre structure proposée avec le SVM pour les occultations est aussi à noter grâce à ces résultats de simulation. Une piste intéressante suite à ces travaux est d'aller plus loin dans l'apprentissage de l'interaction entre plusieurs systèmes.

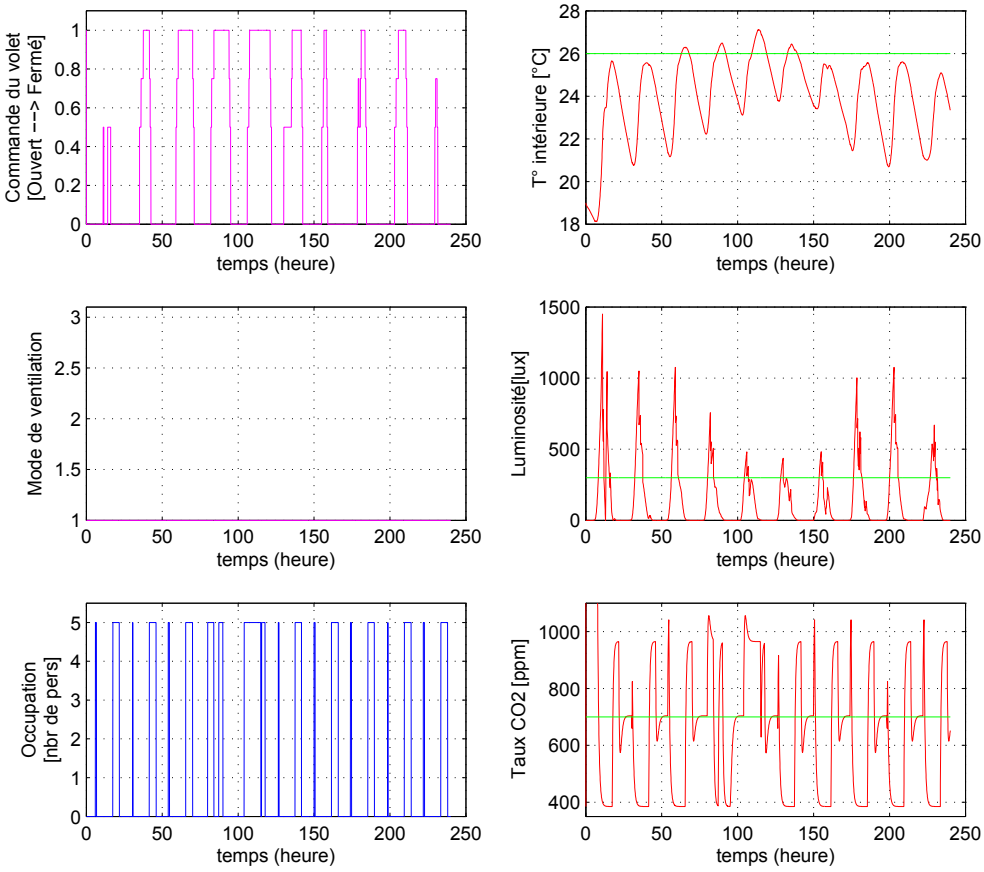

FIGURE *6.5: Différentes mesures à l'intérieur de la maison avec le pilotage des occultations et la ventilation fonctionnant à petite vitesse*

6.3.3.2 Eté

Dans cette partie, les résultats de simulation obtenus en été seront présentés (cf. Figure 6.6). Nous effectuons ici une simulation pendant 6 jours dont le jour de début est le 176e jour.

Il est clair que la protection par des occultations devient insuffisante face à la température extérieure très importante, qui dépasse $30\,^{\circ}\mathrm{C}$ et qui peut même atteindre $35\,^{\circ}\mathrm{C}$ comme dans cette période. De plus, une ouverture de fenêtre n'est pas non plus une bonne idée, ce qui peut dégrader encore plus le confort thermique à cause de la chaleur chaude venant de l'extérieure et qui peut chauffer encore plus l'intérieur.

Ces situations avaient été montrées dans les résultats de simulation avec le contrôleur optimal MPC. Elles ont été retrouvées dans cette partie avec le contrôleur logique, ce qui montre l'efficacité des algorithmes d'apprentissage mis en œuvre. En effet, l'algorithme de l'arbre de décision a permis de bien apprendre les comportements du MPC, même quand ils ne sont pas satisfaisants.

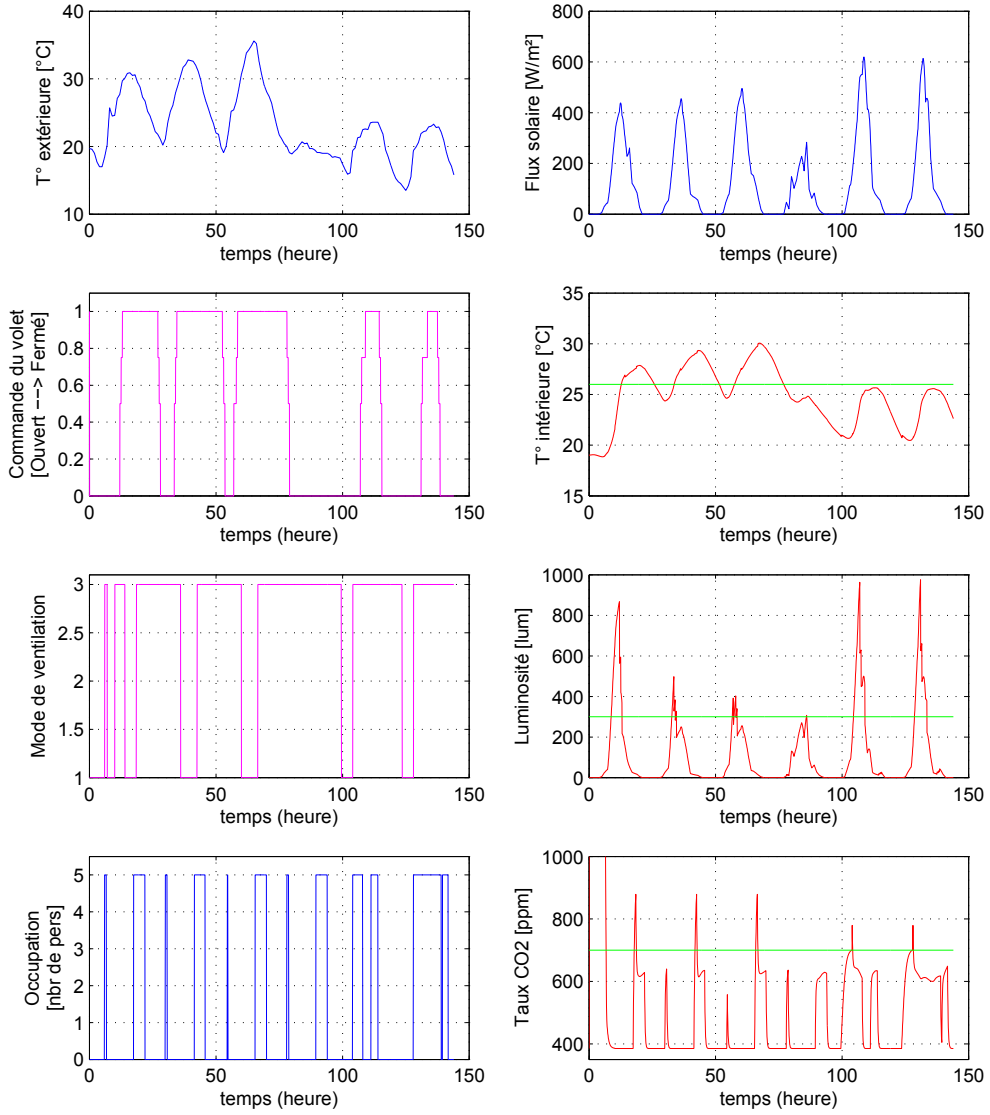

FIGURE *6.6: Comportement du contrôleur logique pendant une période de chaud*

6.4 Conclusion

Ce chapitre présente la deuxième étape de notre méthodologie de développement d'un contrôleur logique, le but est d'apprendre les comportements du contrôleur optimal grâce à des algorithmes d'apprentissage automatique. Nous avons proposé d'étudier ici un cas plus complexe avec le fonctionnement de l'ensemble des occultations et le système de ventilation.

Nous avons montré à travers différentes simulations que les algorithmes comme SVM ou l'arbre de décision sont appropriés à l'apprentissage des comportements du contrôleur MPC. Les différents étapes importantes pour un problème d'apprentissage automatique sont aussi reappliquées. Nous commençons toujours par créer une base de données à partir des simulation avec le MPC. Ensuite, la fonction "Sélection d'attributs" nous permet de bien choisir les mesures les plus utiles à mettre en valeur. Suite à des analyses visuelles de la base de données, nous avons décidé de garder l'approche SVM pour apprendre le volet tandis que l'approche de l'arbre de décision a été choisie pour apprendre le système de ventilation.

En effet, comme montré dans la partie précédente, l'outil SVM propose toujours une structure de contrôleur logique qui est non seulement simple, basé sur deux mesures de flux solaire et de la température intérieure mais aussi efficace, ce qui permet de bien assurer le confort thermique de l'occupant. Quant au système de ventilation, la méthode de l'arbre de décision a montré son efficacité en relevant des comportements marquants du contrôleur optimal MPC et en proposant ensuite des règles pour construire un contrôleur logique.

Quatrième partie

Conclusions et perspectives

Résumé de la thèse, conclusions et perspectives

Résumé

Ce manuscrit a été détaillé dans le but de présenter notre méthodologie pour développer un contrôleur pour les équipements d'un logement. Il s'agit d'une procédure pour construire un contrôleur logique grâce à des règles issues d'un apprentissage des comportements d'un contrôleur optimal de type MPC. Notre méthodologie a été appliquée dans deux cas d'étude : d'abord pour le pilotage d'un volet et ensuite pour contrôler le volet, l'ouvrant et le système de ventilation.

Nous avons essayé, tout au long de ce manuscrit, de respecter l'ordre de présentation de notre procédure qui se déroule en deux grandes étapes, afin de la rendre la plus claire possible.

La première étape consiste à développer un contrôleur optimal MPC. Des modèles de prédiction sont nécessaires pour formuler par la suite un problème d'optimisation avec des critères de conforts de l'occupant et des contraintes de fonctionnement des équipements. La méthode Branch and Bound a été choisie pour résoudre le problème d'optimisation, qui consiste à développer toutes les solutions possibles sous forme d'une arbre et puis à parcourir l'arbre afin de trouver la solution optimale.

Après avoir réussi à mettre en œuvre la commande MPC, la deuxième étape de notre procédure consiste à appliquer des algorithmes d'apprentissage automatique sur la base de données obtenues à partir des simulations du MPC. Différents algorithmes tels que l'arbre de décision, AdaBoost et SVM ont été testés sur notre base de données et qui montrent leurs avantages et leurs inconvénients. Les règles déduites de ces études ont été utilisées par la suite pour construire un contrôleur logique.

Enfin, différentes simulations ont été effectuées afin d'évaluer la performance des contrôleurs et de comparer le résultats obtenus avec différents contrôleurs.

Conclusions

Le contrôleur MPC a montré son efficacité dans le contrôle du bâtiment grâce à sa capacité d'anticipation mais il n'est pas une solution adaptée à l'heure actuelle pour être implémenté dans un produit commercial car il exige des efforts de calcul importants. En revanche, le contrôleur logique développé grâce à notre méthodologie est très simple et il se base uniquement sur des règles logiques et des opérateurs simples. Certes il est moins efficace que le contrôleur optimal, mais les résultats obtenus avec les deux contrôleurs pour les deux cas d'études sont très proches.

Différents défis ont été relevés pendant le développement du contrôleur optimal MPC.

— *Défi 1 : Comment bien modéliser le bâtiment ?*

  En effet, le MPC a l'avantage d'adapter les commandes en fonction des perturbations (conditions météorologiques etc.) et des prédictions du futur comportement du bâtiment, tout cela grâce à un modèle. C'est pourquoi, le premier obstacle est de bien développer

un modèle approprié qui est capable de fournir de bonnes prédictions. L'approche d'identification de type boîte noire a été choisie de par sa simplicité. Par contre, cette approche devient compliquée s'il faut mettre en œuvre plusieurs modèles différents en même temps. Comme les matrices d'états des modèles ne sont pas uniques, il est nécessaire d'exprimer tous les modèles dans une même base. Nous avons réussi à résoudre ce problème en proposant un protocole expérimental pour déterminer les matrices de changement de base.

— *Défi 2 : Comment trouver un compromis des critères ?*
  Une autre difficulté dans la mise en œuvre du MPC est le choix des pondérations dans un problème multicritère. Nous avons proposé une méthode pour bien trouver un compromis entre les différents critères. En effet, nos critères sont très souvent antagonistes et les courbes représentant ces critères évoluent de manière inverse en fonction de la pondération, ce qui permet par la suite de trouver un compromis qui correspond à l'endroit où les courbes se croisent.

— *Défi 3 : Comment résoudre le problème d'optimisation ?*
  Afin de trouver la meilleure commande pour les équipements, nous devons résoudre un problème d'optimisation qui est souvent de grande taille, d'autant plus que les systèmes à contrôler sont nombreux. La méthode de résolution Branch and Bound a montré son efficacité pour résoudre le problème d'optimisation avec les sorties discrètes. Elle propose aussi une technique de découpage de l'arbre pour simplifier la résolution du problème d'optimisation et réduire ainsi le temps de calcul. Par ailleurs, nous avons proposé une technique simple pour pouvoir réduire encore plus l'effort de calcul en réduisant l'espace de recherche. Un autre avantage non négligeable de la méthode Branch and Bound est qu'elle n'est pas trop lourde à être implémentée dans un produit industriel si nous voulons commercialiser notre solution

Quant à l'apprentissage automatique :

— *Défi 4 : Comment construire une base de données ?*
  Les données jouent un rôle important dans le problème d'apprentissage automatique. Nos expériences montrent que nous pouvons réduire la taille de la base de données en choisissant bien la période de simulation. En effet, il faut aussi trouver un équilibre entre la quantité de données à apprendre et la qualité du résultat d'apprentissage. Une quantité trop grande de données peut alourdir le problème d'apprentissage tandis que trop peu de données ne peuvent pas nous apporter des règles logiques pertinentes. Nous avons montré que pour le cas du pilotage des occultations seules, une simulation pendant un mois de mi-saison est suffisante car nous trouvons des données très variées pendant cette période. De même pour le cas de la gestion de la ventilation, il est possible d'apprendre en mi-saison où il y a une variation remarquable de conditions météorologiques, surtout en France.

— *Défi 5 : Comment mettre en œuvre les techniques d'apprentissage ?*
  Pour bien construire un contrôleur logique avec des règles obtenues grâce à un apprentissage automatique, il est primordial de comprendre et de bien choisir les mesures à prendre en compte pour construire la base de données utilisée pour l'apprentissage. Nous avons montré à travers des études de "feature selection" et des simulations qu'une mesure redondante peut affaiblir la capacité d'apprentissage et réduire ainsi la performance du contrôleur

appris.
Plusieurs techniques d'apprentissage automatiques existent dans la littérature. Il est important de pouvoir tester et comparer pour savoir quel outil est le plus approprié pour le problème étudié. L'expérience montre qu'aucun outil d'apprentissage n'est approprié à tout problème d'exploration de données. L'apprentissage universel semble pour l'instant inatteignable.
Nous avons fait aussi le choix d'apprendre chaque système individuellement au lieu d'un apprentissage global de tous les systèmes. Non seulement cela permet de simplifier le problème d'apprentissage, mais aussi cela est adapté pour mettre en œuvre une structure évolutive qui permet d'ajouter et supprimer facilement différents systèmes à contrôler, tout en affectant peu la performance de l'ensemble des systèmes qui sont en place.

— *Défi 6 : Comment construire un contrôleur logique ?*
A partir des règles de décision fournies par les algorithmes d'apprentissage automatique, la question est maintenant de savoir comment construire le contrôleur logique. Pour le cas de la gestion des volets, nous avons proposé une structure de machine d'état pour gérer les règles obtenues avec SVM. Nous avons vu aussi l'intérêt de la méthode de l'arbre de décision qui peut nous fournir directement une structure interprétable pour aider la décision.

Perspectives

Plusieurs perspectives peuvent être envisagées à partir de ces travaux de thèse :

— Un premier axe d'amélioration est la prise en compte des différents critères dans le problème d'apprentissage. En effet, nous avons adopté une représentation linéaire pour nos critères de confort thermique, visuel, CO_2 etc. afin de simplifier notre problème d'optimisation. Les différents conforts pour l'occupant pourraient être formulés de manière plus sophistiqués, comme nous avons cité par exemple le confort Brager pour la thermique, ou encore avec les indices de confort thermique comme le PMV et PPD [70, 71], etc. De plus, la traduction des autres conforts doit aussi être étudiée, particulièrement le confort sonore ou l'impact du courant d'air sur le confort thermique. Comment ces facteurs influencent la décision du pilotage des occultations ou encore la gestion de la ventilation doit être étudié.

— Notre contrôleur optimal MPC pour la gestion de la ventilation est basé sur les mesures de la température opérative qui est la moyenne de la température de l'enveloppe et celle de l'air. C'est pourquoi, l'effet d'une ouverture de fenêtre n'a pas été relevé dans nos études. Pourtant, suite à l'ouverture d'un ouvrant, la température de l'air peut monter très significativement et causer une sensation d'inconfort chez l'occupant tandis que la température opérative peut ne pas prendre en compte ce phénomène. Une piste d'amélioration de ces travaux de thèse est donc d'étudier le contrôle des ouvrants avec une mesure de la température de l'air.

— Une autre perspective est la formulation du problème quand il y a plusieurs systèmes à contrôler, qui ont chacun un horizon de prédiction différent. En effet, comment formuler notre problème de contrôle optimal dans ces cas d'étude ? Que devient-il le problème d'apprentissage automatique ?

— Il est important d'insister ici que les règles de logiques obtenues avec les algorithmes d'apprentissage ne sont que les règles les plus pertinentes. C'est pourquoi, les phénomènes particuliers ne sont pas dans la liste des règles et ils peuvent être sous-estimées. Comment améliorer la qualité du résultat d'apprentissage automatique qui peut faire sortir des phénomènes rares sans devoir être confronté au problème d'"over-fitting" ? C'est une piste intéressante pour continuer ces travaux de thèse.
— Aspect expérimental : nos travaux se sont limités à des validations par simulation, montrant de belles performances, malgré une capacité de calcul limitée. La prochaine étape est donc de tester ces solutions sur le terrain, dans de vraies maisons, avec de vrai occupants.

Table des figures

1.1 Six facteurs affectent le confort thermique [http ://labo-energetic.eu] 4
1.2 Différents plages de confort hygrothermique [http ://www.energieplus-lesite.be] . . 5
1.3 Proposition d'une structuration abstraite du fonctionnement de l'ensemble des systèmes dans l'habitat 7
1.4 Schéma représentant différents étapes de notre méthodologie 8
1.5 Schéma des interactions pour le pilotage de l'ensemble des occultations, de la ventilation et de l'ouvrant 9

2.1 Différents éléments importants pour la formulation d'un problème de contrôle prédictif. Nous y trouvons les objectifs et les contraintes, les modèles de prédiction ainsi qu'un problème d'optimisation suivi d'une méthode de résolution 16
2.2 Principe de fonctionnement de la commande prédictive MPC 18
2.3 Explication du principe de fonctionnement de la méthode Branch & Bound 18
2.4 Principaux éléments d'un problème d'apprentissage automatique 20
2.5 Exemple de présentation en deux dimensions des données enregistrées. Les décisions sont "fermer le volet" et "ouvrir le volet". Les attributs sont la température intérieure et le flux solaire 21
2.6 Principe du découpage d'algorithme d'arbre de décision avec notre exemple pour le pilotage des volets 22
2.7 L'arbre de décision construit à partir des résultats du découpage précédent, avec notre exemple pour le pilotage des volets 23
2.8 Exemple de l'application de la méthode d'AdaBoost. La décision finale est la somme du produit de toutes les classificateurs simples et leur pondération 24
2.9 Représentation graphique du classificateur final obtenu avec la méthode d'Adaboost 25
2.10 L'hyperplan séparateur obtenu avec le SVM. Le principe de fonctionnement du SVM se base sur les vecteurs supports et la marge maximale 26
2.11 Interface de bienvenue de Weka 28
2.12 La description du bâtiment avec le fichier XML 30
2.13 Plan de la maison 31

3.1 Le schéma du MPC pour la gestion des occultations 37
3.2 Performance du modèle thermique, résultats de comparaison entre la courbe réelle et la courbe prédite sur un horizon de prédiction de 10 heures 39
3.3 Diagramme de Brager 40
3.4 L'influence de la pondération sur les critères 42
3.5 L'influence de la pondération sur le nombre de mouvements 42
3.6 Relation entre la violation de la température et l'horizon de prédiction 43
3.7 Conditions météorologiques avec les mesures de température extérieure et le flux solaire. Le confort thermique des occupants s'est dégradé avec la température intérieure qui dépasse le seuil de confort fixé à $26\,^{\circ}\mathrm{C}$ 44

3.8 Résultats obtenus avec le MPC. Les conditions météorologiques sont parfaitement connues à l'avance. 45
3.9 Résultats obtenus avec le MPC. Les conditions météorologiques ne sont pas connues à l'avance, elles sont bloquées sur tout l'horizon de prédiction à la valeur constante 46
3.10 Résultats obtenus avec un contrôleur classique P 46

4.1 Conditions météorologiques en mars de la ville Vaulx-en-Velin (France) et la température intérieure sans protection des volets 51
4.2 Résultats obtenus avec le MPC pour la consigne à $24\,°\mathrm{C}$ et à $26\,°\mathrm{C}$ 52
4.3 Résultat obtenu avec la méthode C4.5 52
4.4 Présentation du contrôleur simplifié SVM sous forme de machine d'état 53
4.5 Résultats de simulation obtenus avec les différents contrôleurs simplifiés 54
4.6 Comparaison de trois méthodes selon différents critères 54
4.7 Exemple de $L_{(075_05)}$ obtenu par SVM. La décision est basée sur des mesures du flux solaire et de la température extérieure 57
4.8 Résultats de simulation en juin avec des conditions météorologiques différentes et une maison différente. La consigne est mise à $26°\mathrm{C}$ 61
4.9 Résultats de simulation en juin avec des conditions météorologiques différentes et une maison différente. La consigne est mise à $25°\mathrm{C}$. Les paramètres du contrôleur pour cette consigne ont été adaptés à partir des paramètres du contrôleur de base pour la consigne à $26°\mathrm{C}$ 62

5.1 L'étiquette sanitaire des produits de construction et de décoration 67
5.2 Schéma MPC pour la gestion des occultations et du système de ventilation 68
5.3 Créneaux de commande pour la génération de données 71
5.4 Enregistrement des couples d'états au moment du changement de consigne . . . 71
5.5 La prédiction avec le changement de base, comparée à la courbe réelles, 187 ^{e} jour 72
5.6 Exemple du courant d'air créé par une ouverture individuelle de fenêtre 72
5.7 Comparaison d'une prédiction de température intérieure 7.5 heures en avance avec un fonctionnement mixte de différentes positions du volet et différents modes de ventilation 74
5.8 Plan de la distribution d'air de la maison 74
5.9 L'évolution de la violation du confort thermique et du confort CO_2 en fonction de la valeur de la pondération du confort CO_2 78
5.10 L'évolution de la consommation d'électricité normalisée en fonction de la pondération sur la consommation 79
5.11 Augmentation du pas d'échantillonnage de commande pour calculer le critère afin réduire l'effort de calcul 80
5.12 Conditions météorologiques et mesures de température intérieure et du CO_2 dans la pièce avec le volet restant ouvert et la ventilation fonctionnant à petite vitesse . . 82
5.13 Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations pendant une période de mi-saison 83

5.14 Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations 84
5.15 Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations dans le cas où il y a 2 occupants 85
5.16 Résultats de simulation du contrôle optimal de l'ensemble du système de ventilation et des occultations pendant une période estivale 86

6.1 Principales données météorologiques de la période de simulation 90
6.2 L'arbre de décision obtenu par l'apprentissage d'un contrôleur optimal pour le système de ventilation 94
6.3 Résultats de simulation obtenus avec le fonctionnement de l'ensemble des occultations et le système de ventilation 96
6.4 Résultats de simulation obtenus avec le fonctionnement de l'ensemble des occultations et le système de ventilation. Les paramètres du contrôleur logique pour le pilotage des volets viennent de l'étude précédente 98
6.5 Différentes mesures à l'intérieur de la maison avec le pilotage des occultations et la ventilation fonctionnant à petite vitesse 99
6.6 Comportement du contrôleur logique pendant une période de chaud 100

Liste des tableaux

3.1 Tableau de consommation de différents contrôleurs 47

4.1 La valeur du paramètre w des hyperplans $H_{(1_075)}$, $H_{(075_05)}$, $H_{(05_0)}$ 55
4.2 Comparaison de la performance de différents contrôleurs avec la consigne à $26\,^{\circ}\mathrm{C}$ en mars 56
4.3 Comparaison de la pente α obtenue par SVM selon la période d'apprentissage . . 57
4.4 Comparaison du paramètre de seuil β obtenu par SVM selon la période d'apprentissage 57
4.5 Comparaison de la pente α et le paramètre β des résultats obtenus par un apprentissage en mars et en juillet 58
4.6 Comparaison de LC2-M et LC2-J selon différents critères avec une simulation en mars pour une consigne à $26\,^{\circ}\mathrm{C}$ 58
4.7 Comparaison de LC2-M et LC2-J selon différents critères avec une simulation en juillet pour une consigne à $26\,^{\circ}\mathrm{C}$ 58
4.8 Comparaison de la valeur de la pente α et du coefficient β obtenus grâce au SVM pour la consigne à $26\,^{\circ}\mathrm{C}$ et celle à $24\,^{\circ}\mathrm{C}$ 59
4.9 Comparaison de la performance de contrôleurs simplifiés appris par différentes méthodes avec le contrôleur optimal avec la consigne à $24\,^{\circ}\mathrm{C}$ 60
4.10 Résultats d'une longue simulation (d'avril au juin) avec la consigne à $26\,^{\circ}\mathrm{C}$ 60

6.1 La valeur du paramètre w des hyperplans $H_{(1_075)}$, $H_{(075_05)}$, $H_{(05_0)}$ 92
6.2 La valeur du paramètre w des hyperplans $H_{(v1_v3)}$ qui sépare la décision de la ventilation mécanique à petite vitesse et celle de la ventilation naturelle 92
6.3 Valeur de la pente α et le paramètre β obtenus par un apprentissage avec le SVM 93
6.4 Comparaison de la pente α et le paramètre β des résultats obtenus par un apprentissage dans les deux cas d'étude 97

Bibliographie

[1] S. Kirchner, J.-F. Arenes, C. Cochet, M. Derbez, C. Duboudin, P. Elias, A. Gregoire, B. Jédor, J.-P. Lucas, N. Pasquier et al. *État de la qualité de l'air dans les logements français*. Environnement, Risques & Santé 6, 259 (2007).

[2] R. Fauconnier. *L'action de l'humidité de l'air sur la santé dans les bâtiments tertiaires*. Chauffage, ventilation, conditionnement 68, 57 (1992).

[3] D. Molle and P.-M. Patry. *RT 2012 et RT existant : Réglementation thermique et efficacité énergétique-Construction et rénovation* (Editions Eyrolles, 2013).

[4] P. Michel. *Conduite des systèmes climatiques* (Ed. Techniques Ingénieur, 2002).

[5] C. E. Garcia, D. M. Prett and M. Morari. *Model predictive control : theory and practice—a survey*. Automatica 25, 335 (1989).

[6] A. Afram and F. Janabi-Sharifi. *Theory and applications of hvac control systems–a review of model predictive control (mpc)*. Building and Environment 72, 343 (2014).

[7] J. Širokỳ, F. Oldewurtel, J. Cigler and S. Prívara. *Experimental analysis of model predictive control for an energy efficient building heating system*. Applied Energy 88, 3079 (2011).

[8] S. Privara, J. Širokỳ, L. Ferkl and J. Cigler. *Model predictive control of a building heating system : The first experience*. Energy and Buildings 43, 564 (2011).

[9] S. Teleke, M. E. Baran, S. Bhattacharya and A. Q. Huang. *Rule-based control of battery energy storage for dispatching intermittent renewable sources*. Sustainable Energy, IEEE Transactions on 1, 117 (2010).

[10] M. Gwerder, D. Gyalistras, F. Oldewurtel, B. Lehmann, K. Wirth, V. Stauch and J. Tödtli. *Potential assessment of rule-based control for integrated room automation*. In 10th *REHVA World Congress, "Sustainable Energy Use in Buildings" - Clima 2010, Antalya* (2010).

[11] S. J. Qin and T. A. Badgwell. *An overview of industrial model predictive control technology*. In *AIChE Symposium Series* (New York, NY : American Institute of Chemical Engineers, 1971-c2002., 1997), volume 93, pp. 232–256.

[12] S. J. Qin and T. A. Badgwell. *A survey of industrial model predictive control technology*. Control engineering practice 11, 733 (2003).

[13] D. W. Clarke, C. Mohtadi and P. Tuffs. *Generalized predictive control—part i. the basic algorithm*. Automatica 23, 137 (1987).

[14] M. V. Kothare, V. Balakrishnan and M. Morari. *Robust constrained model predictive control using linear matrix inequalities*. Automatica 32, 1361 (1996).

[15] S. J. Qin and T. A. Badgwell. *An overview of nonlinear model predictive control applications*. In *Nonlinear model predictive control* (Springer, 2000), pp. 369–392.

[16] E. F. Camacho, D. R. Ramírez, D. Limón, D. Muñoz de la Peña and T. Alamo. *Model predictive control techniques for hybrid systems*. Annual reviews in control 34, 21 (2010).

[17] M. Morari and J. H. Lee. *Model predictive control : past, present and future*. Computers & Chemical Engineering 23, 667 (1999).

[18] F. Borrelli, A. Bemporad, M. Fodor and D. Hrovat. *An mpc/hybrid system approach to traction control*. Control Systems Technology, IEEE Transactions on 14, 541 (2006).

[19] L. A. Wolsey and G. L. Nemhauser. *Integer and combinatorial optimization* (John Wiley & Sons, 2014).

[20] K. Marriott and P. J. Stuckey. *Programming with constraints : an introduction* (MIT press, 1998).

[21] I. I. Cplex. *12.2 user's manual* (2010).

[22] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. Savelsbergh and P. H. Vance. *Branch-and-price : Column generation for solving huge integer programs*. Operations research 46, 316 (1998).

[23] E. L. Lawler and D. E. Wood. *Branch-and-bound methods : A survey*. Operations research 14, 699 (1966).

[24] P. Ferreira, A. Ruano, S. Silva and E. Conceição. *Neural networks based predictive control for thermal comfort and energy savings in public buildings*. Energy and Buildings 55, 238 (2012).

[25] J. S. L.F. Mendonça, C.A. Silva. *Branch-and-bound optimization in multivariable model predictive control* (2000).

[26] Y. Ma, F. Borrelli, B. Hencey, A. Packard and S. Bortoff. *Model predictive control of thermal energy storage in building cooling systems*. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on* (IEEE, 2009), pp. 392–397.

[27] E. Alpaydin. *Introduction to machine learning* (MIT press, 2014).

[28] P. Langley and H. A. Simon. *Applications of machine learning and rule induction*. Communications of the ACM 38, 54 (1995).

[29] G. De'ath and K. E. Fabricius. *Classification and regression trees : a powerful yet simple technique for ecological data analysis*. Ecology 81, 3178 (2000).

[30] J. R. Quinlan. *C4. 5 : programs for machine learning* (Elsevier, 2014).

[31] Y. Freund, R. Schapire and N. Abe. *A short introduction to boosting*. Journal-Japanese Society For Artificial Intelligence 14, 1612 (1999).

[32] Y. Freund and R. E. Schapire. *A decision-theoretic generalization of on-line learning and an application to boosting*. Journal of Computer and System Sciences 55, 119 (1997).

[33] V. V. Carinna Cortes. *Support-vector networks*. Machine Learning, 20, 273-297 (1995) (1995).

[34] A. J. Smola and B. Schölkopf. *A tutorial on support vector regression*. Statistics and computing 14, 199 (2004).

[35] C. J. Burges. *A tutorial on support vector machines for pattern recognition*. Data mining and knowledge discovery 2, 121 (1998).

[36] O. Ivanciuc. *Applications of support vector machines in chemistry*. Reviews in computational chemistry 23, 291 (2007).

[37] S. R. Gunn. *Support vector machines for classification and regression*. ISIS technical report 14 (1998).

[38] L. Kaufman and F. Broeckx. *An algorithm for the quadratic assignment problem using bender's decomposition*. European Journal of Operational Research 2, 207 (1978).

[39] M. Frank and P. Wolfe. *An algorithm for quadratic programming*. Naval research logistics quarterly 3, 95 (1956).

[40] C.-C. Chang and C.-J. Lin. *Libsvm : A library for support vector machines*. ACM Transactions on Intelligent Systems and Technology (TIST) 2, 27 (2011).

[41] D. M. Hawkins. *The problem of overfitting*. Journal of chemical information and computer sciences 44, 1 (2004).

[42] K. Leahy. *The overfitting problem in perspective*. AI Expert 9, 35 (1994).

[43] T. Dietterich. *Overfitting and undercomputing in machine learning*. ACM computing surveys (CSUR) 27, 326 (1995).

[44] C. Schaffer. *Overfitting avoidance as bias*. Machine learning 10, 153 (1993).

[45] S. Shah and P. S. Sastry. *New algorithms for learning and pruning oblique decision trees*. Systems, Man, and Cybernetics, Part C : Applications and Reviews, IEEE Transactions on 29, 494 (1999).

[46] J. R. Quinlan. *Bagging, boosting, and c4. 5*. In *AAAI/IAAI, Vol. 1* (1996), pp. 725–730.

[47] I. Guyon and A. Elisseeff. *An introduction to variable and feature selection*. The Journal of Machine Learning Research 3, 1157 (2003).

[48] K. Kira and L. A. Rendell. *The feature selection problem : Traditional methods and a new algorithm*. In *AAAI* (1992), volume 2, pp. 129–134.

[49] K. Kira and L. A. Rendell. *A practical approach to feature selection*. In *Proceedings of the ninth international workshop on Machine learning* (1992), pp. 249–256.

[50] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten. *The weka data mining software : an update*. ACM SIGKDD explorations newsletter 11, 10 (2009).

[51] U. of Wisconsin-Madison. Solar Energy Laboratory and S. A. Klein. *TRNSYS, a transient system simulation program* (Solar Energy Laborataory, University of Wisconsin–Madison, 1979).

[52] D. B. Crawley, L. K. Lawrie, C. O. Pedersen and F. C. Winkelmann. *Energy plus : energy simulation program*. ASHRAE journal 42, 49 (2000).

[53] A. Husaunndee, R. Lahrech, H. Vaezi-Nejad and J. Visier. *Simbad : A simulation toolbox for the design and test of hvac control systems*. In *Proceedings of the 5th international IBPSA conference, Prague, Czech Republic* (1997), volume 2, pp. 269–276.

[54] A. Lefort, R. Bourdais, G. Ansanay-Alex and H. Guéguen. *Hierarchical control method applied to energy management of a residential house*. Energy and Buildings 64, 53 (2013).

[55] P. Morosan, R. Bourdais, D. Dumur and J. Buisson. *Distributed model predictive control for building temperature regulation*. In *American Control Conference (ACC), 2010* (IEEE, 2010), pp. 3174–3179.

[56] *Entpe (http ://idmp.entpe.fr/vaulx/stafr.htm)*.

[57] A. Tzempelikos and A. K. Athienitis. *The impact of shading design and control on building cooling and lighting demand*. Solar Energy 81, 369 (2007).

[58] M. V. Nielsen, S. Svendsen and L. B. Jensen. *Quantifying the potential of automated dynamic solar shading in office buildings through integrated simulations of energy and daylight*. Solar Energy 85, 757 (2011).

[59] G. Kim, H. S. Lim, T. S. Lim, L. Schaefer and J. T. Kim. *Comparative advantage of an exterior shading device in thermal performance for residential buildings*. Energy and Buildings 46, 105 (2012).

[60] G. Brager and R. de Dear. *A standard for natural ventilation*. ASHRAE journal (2000).

[61] P. May-Ostendorp, G. P. Henze, C. D. Corbin, B. Rajagopalan and C. Felsmann. *Model-predictive control of mixed-mode buildings with rule extraction*. Building and Environment 46, 428 (2011).

[62] A. Domahidi, F. Ullmann, M. Morari and C. N. Jones. *Learning decision rules for energy efficient building control*. Journal of Process Control (2014).

[63] C.-W. Hsu and C.-J. Lin. *A comparison of methods for multiclass support vector machines*. Neural Networks, IEEE Transactions on 13, 415 (2002).

[64] L. Mølhave. *Indoor climate, air pollution, and human comfort.* Journal of exposure analysis and environmental epidemiology 1, 63 (1991).

[65] F. C. supérieur d'hygiène publique de France. Section de l'évaluation des risques de l'environnement sur la santé, Y. Le Moullec and F. Squinazi. *Pollution atmosphérique à l'intérieur des bâtiments : sources, expositions et risques sanitaires : analyse bibliographique des études françaises, 1983-1993* (Tec et doc, 1996).

[66] C.-A. Roulet. *Santé et qualité de l'environnement intérieur dans les bâtiments*, volume 22 (PPUR presses polytechniques, 2004).

[67] F. Squinazi. *La pollution de l'air à l'intérieur des bâtiments (allergènes exclus)*. Revue française d'allergologie et d'immunologie clinique 42, 248 (2002).

[68] P. Wargocki, D. P. Wyon, J. Sundell, G. Clausen and P. Fanger. *The effects of outdoor air supply rate in an office on perceived air quality, sick building syndrome (sbs) symptoms and productivity*. Indoor air 10, 222 (2000).

[69] T. S. Larsen and P. Heiselberg. *Single-sided natural ventilation driven by wind pressure and temperature difference*. Energy and Buildings 40, 1031 (2008).

[70] O. internationale de normalisation. *Moderate Thermal Environnements : Determination of the PMV and PPD Indices and Specification of the Conditions for Thermal Comfort* (International Organization for Standardization, 1984).

[71] E. Iso. *7730 : 2005 :"ergonomics of the thermal environment–analytical determination and interpretation of thermal comfort using calculation of the pmv and ppd indices and local thermal comfort criteria"*. International Organization for Standardisation, Geneva (2005).