



HAL
open science

Méthode et outils pour l'identification de défauts des bâtiments connectés performants

Rozenn Josse

► **To cite this version:**

Rozenn Josse. Méthode et outils pour l'identification de défauts des bâtiments connectés performants. Energie électrique. Université Grenoble Alpes, 2017. Français. NNT : 2017GREAT074 . tel-01743757

HAL Id: tel-01743757

<https://theses.hal.science/tel-01743757>

Submitted on 26 Mar 2018

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.

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : **Génie électrique**

Arrêté ministériel : 25 mai 2016

Présentée par

Rozenn JOSSE

Thèse dirigée par **Frédéric WURTZ**, directeur de recherche,
CNRS, et
codirigée par **Etienne WURTZ**, directeur de recherche, **CEA-
INES**

préparée au sein du **Laboratoire de Génie Electrique de
Grenoble (G2ELAB)**

dans l'**École Doctorale d'Electronique, Electrotechnique,
Automatique et Traitement du Signal (EEATS)**

Méthode et outils pour l'identification de défauts des bâtiments connectés performants

Thèse soutenue publiquement le **13 novembre 2017**,
devant le jury composé de :

M. Jean-Jacques ROUX

Professeur, INSA de Lyon, Rapporteur

M. Stéphane LASSUE

Professeur, Université d'Artois, Rapporteur

Mme Monika WOLOSZYN

Professeur, Université de Savoie Mont Blanc, Présidente du jury

M. Laurent MORA

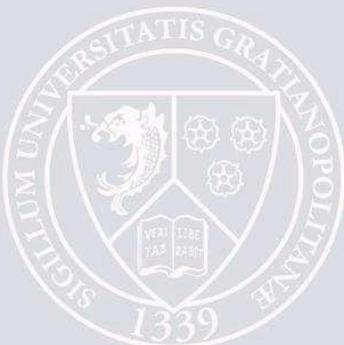
Maitre de conférences, IUT Bordeaux, Examineur

M. Frédéric WURTZ

Directeur de recherche, G2ELAB, Directeur de thèse

M. Etienne WURTZ

Directeur de recherche, CEA-INES, Co-Directeur de thèse



Résumé



Ces travaux de thèse portent sur le développement d'une nouvelle méthodologie pour l'identification de défauts de bâtiments performants et connectés afin d'aider à la garantie de performances. Nous avons dans un premier temps resitué nos travaux dans le contexte énergétique actuel en montrant le rôle majeur des bâtiments dans la réduction des consommations énergétiques. Nous avons ensuite présenté notre méthodologie en argumentant sur les techniques à utiliser avant d'effectuer un choix final. Cette méthodologie se compose de deux blocs principaux : le premier vise à réduire les incertitudes liées à l'occupant et à l'environnement et le second étudie l'écart entre la simulation et la mesure par une analyse de sensibilité couplée à un algorithme bayésien. Nous l'avons ensuite implémentée dans un outil que nous avons nommé REFATEC. Nous avons alors soumis notre méthodologie à différents tests dans des conditions idéales afin d'éprouver sa précision et son temps d'exécution. Cette étape a montré que la méthodologie est efficace mais montre quelques faiblesses dans le cas d'une saison estivale ou d'un défaut très localisé. Enfin, nous l'avons mise en situation face à un cas réel afin de traiter les nombreuses questions que soulèvent l'utilisation de mesures in-situ dans la perspective de la garantie de performances et de la détection de défauts, avec notamment la fiabilité des mesures et les incertitudes encore nombreuses qui doivent être traitées.

Mots clés : *Identification de défauts, bâtiment performant et connecté, analyse de sensibilité, algorithme bayésien, garantie de performances.*

Abstract



This thesis deals with the development of a new methodology for fault detection within smart high-performance buildings helping the performance guarantee. We first have placed our work in the current energy context by focusing on the major role of buildings in the decrease of energy consumption. Then we introduced our methodology and we argued about various techniques that could be used before making a choice. This methodology is made up of two main parts : the former reduces the uncertainties due to the occupant and the environment and the latter studies the gap between simulation and measurements thanks to a sensitivity analysis coupled with a bayesian algorithm. Then we implemented it within a tool that we named REFATEC. We carried out various tests in controlled conditions in order to evaluate its precision and its calculation time. This step showed that our methodology is effective but it has some difficulties when the studied period is during summer or when the faults are very located. is a very located fault. Eventually we confronted our methodology to a real case where we faced numerous questions that appear when dealing with measurements, especially their reliability and the uncertainties that still need to be taken care of, in the perspective of performance guarantee and fault detection.

Key words : *Fault detection, smart high-performance building, sensitivity analysis, bayesian algorithm, performance guarantee.*

Table des matières



Remerciements	xiv
Table des figures	xix
Liste des tableaux	xxi
Nomenclature	xxvi
Introduction	1
I Vers la garantie de performances	3
1 Un contexte énergétique particulier : une prise de conscience mais des consommations toujours élevées	6
1.1 Le rôle du bâtiment pour atteindre les objectifs environnementaux	6
1.2 Le bâtiment et l'électricité : un consommateur producteur	7
2 Les enjeux du bâtiment connecté	9
2.1 Présentation et définition d'un bâtiment connecté pour ces travaux de thèse	9
2.2 Les développements académiques	11
2.3 Le marché des offres grands publics	11
2.4 Le bâtiment intelligent : un atout pour la transition énergétique	12
3 La ou plutôt les garanties de performances	13
3.1 Définition de la garantie de performances (GP)	13
3.1.1 Garantie de Performance Énergétique Intrinsèque (GPEI)	14

3.1.2	Garantie de Résultats Energétiques (GRE) sur l'usage	15
3.2	Plans mis en oeuvre pour y répondre.	16
3.2.1	Contrat de Performance Energétique (CPE)	16
3.2.2	International Performance Measurement & Verification Protocol (IPMVP)	17
3.2.3	ASHRAE : Guideline 14-2002	19
4	Besoins ressentis et notre proposition pour y répondre	20
II Une nouvelle méthode pour l'identification des défauts		23
1	Introduction	26
2	L'identification de défauts dans le monde du bâtiment	26
2.1	Etat de l'art sur l'identification de défauts	26
2.1.1	Méthodes d'identification de défauts en cas de présence des occupants	27
2.1.2	Méthodes d'identification de défauts en cas d'absence des occupants	27
2.2	Problèmes liés à ces méthodes	29
3	Principe de la méthode proposée pour l'identification de défauts	30
3.1	Raisonnement pour l'établissement de la méthode	30
3.1.1	Problématique : un écart important entre la phase de conception et celle d'habitation d'un bâtiment	30
3.1.2	Facteurs influençant l'écart entre les deux phases	30
3.1.3	Réduire le nombre de paramètres à étudier	31
3.1.4	Retrouver la cause de l'écart	32
3.2	Description de l'architecture de la méthode	32
3.2.1	Présentation générale	32
3.2.2	Premier bloc : traitement des mesures	32
3.2.3	Second bloc : comparaison des mesures avec les sorties de la simulation	33
4	Choix des techniques pour l'application de la méthodologie	34
4.1	Identification des périodes d'absence des occupants	34
4.1.1	Etat de l'art de l'identification de la présence des occupants	34

4.1.2	Choix retenu : analyse des consommations	35
4.2	Prise en compte de l'environnement extérieur et des charges internes	37
4.2.1	Etat de l'art	37
4.2.2	Choix retenu : création d'un nouveau scénario et d'un nouveau fichier météorologique	39
4.3	Calcul de l'écart entre mesures et simulation : le choix de la RMSE	39
4.4	Identification des facteurs les plus influents : le choix de la méthode de Morris	40
4.4.1	Etat de l'art des différentes analyses de sensibilité	40
4.4.2	Choix pour la réduction du champ d'études : La méthode de Morris	42
4.5	Identification de défauts : le choix de l'algorithme bayésien ABCPMC	45
4.5.1	Etat de l'art des algorithmes bayésiens	45
4.5.2	Choix pour retrouver les valeurs : l'algorithme ABCPMC	52
5	Présentation de l'outil REFATEC (REmote FAult deTECTion)	52
5.1	Besoins pour la mise en place de l'outil	52
5.2	Architecture de programmation	53
5.2.1	Traitement des mesures	53
5.2.2	Comparaison entre mesure et simulation	54
5.3	Récapitulatif des développements « outil »	57
6	Conclusion	58
 III Présentation, tests et affinage de la méthodologie par simulation de défauts virtuels		 61
1	Introduction	65
2	Présentation du modèle et des hypothèses de simulation prises en compte	66
2.1	Architecture de la maison	66
2.2	Scénarios pris en compte	67
2.3	Défauts à étudier	68

3 Etude de deux indicateurs pour l'identification de défaut	68
3.1 Etude de l'écart de température intérieure entre les deux modèles	69
3.1.1 Fonctionnement des méthodes utilisées appliqué à l'écart de températures	69
3.1.2 Application à la période hivernale	72
3.1.3 Application à la période estivale.	74
3.2 Etude de l'écart de consommation électrique due au chauffage entre les deux modèles sur la période hivernale	78
3.2.1 Fonctionnement des méthodes utilisées appliquées à l'écart de consommation	78
3.2.2 Résultat de la méthode de Morris	79
3.2.3 Résultat de l'algorithme ABCPMC	80
3.3 Conclusion sur les indicateurs et l'impact de la saison sur les résultats	81
4 Etude de l'aspect temporel : optimisation du temps de calcul	81
4.1 Durée de simulation optimale	81
4.2 Durée d'absence optimale	83
4.3 Conclusion sur l'échelle temporelle	85
5 Etude de l'aspect spatial : à quelle échelle arrive-t-on à identifier un défaut?	86
5.1 Défaut minimal détectable par la méthode	86
5.2 Rôle de la durée de l'absence sur la détectabilité du défaut.	87
5.2.1 Evolution de l'épaisseur retrouvée selon la durée d'absence et l'épaisseur cible	87
5.2.2 Evolution de l'écart absolu à l'épaisseur cible selon la durée d'absence	89
5.2.3 Conclusion sur le rôle de la durée d'absence sur la détectabilité d'un défaut	91
5.3 Détection d'un défaut global	92
5.3.1 Résultat de l'analyse de Morris dans le cas d'un défaut global	92
5.3.2 Résultat de l'algorithme ABCPMC dans le cas d'un défaut global	93
5.4 Détection d'un défaut local à l'échelle d'une pièce	93
5.4.1 Mise en place de l'étude	93

5.4.2	Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une pièce	94
5.4.3	Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une pièce	95
5.5	Détection d'un défaut local à l'échelle d'une paroi.	95
5.5.1	Mise en place de l'étude	95
5.5.2	Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une paroi	96
5.5.3	Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une paroi	99
5.6	Conclusion : quelles limites présente notre méthode en terme d'échelle spatiale? . .	100
6	Typologie de défauts détectables	101
6.1	Défauts introduits	101
6.2	Analyse de Morris pour chaque défaut	102
6.2.1	Défauts repérables.	102
6.2.2	Défauts non repérables	104
6.3	Lien entre écart de températures et réparabilité avec l'analyse de Morris	107
7	Conclusion	109
IV	Vers une première mise en situation de l'approche proposée	113
1	Introduction	116
2	Choix d'un cas d'études	116
2.1	Maison POS-A : cas théorique valide pour une étude réelle?	116
2.2	Maison Ostwald : plus adéquat?	117
2.2.1	Présentation	117
2.2.2	Plusieurs défauts repérés sur site	118
2.2.3	Choix du défaut à repérer par notre approche	120
3	Multitude de données disponibles	121
3.1	Sélection des mesures disponibles et adéquates pour l'application de la méthodologie	121

3.2	Données disponibles... et fiables ?	121
4	Application de l'approche proposée et « points relevés »	122
4.1	Analyse des consommations pour l'identification des zones d'absences des occupants	122
4.2	Diminution des incertitudes par la mise à jour des scénarios d'électricité et du fichier météorologique grâce aux mesures	123
4.2.1	De nouveaux scénarios de consommation d'électricité	123
4.2.2	Un nouveau fichier météorologique pour le modèle	124
4.3	Etude de l'écart de température intérieure entre le modèle initial, le modèle mis à jour et la mesure	125
4.4	Analyse de sensibilité sur l'écart de température	128
5	Conclusion de l'application de la méthodologie à un cas réel	132
	Conclusion	135
	Bibliographie	139
	Annexes	146
A	Options IPMVP	147
B	Présentation de l'outil REFATEC	151
1	Premier bloc : Traitement des mesures	152
1.1	Fonctionnement	152
1.2	Code	152
2	Second bloc : Comparaison mesure et simulation	160
2.1	Fonctionnement	160
2.2	Code	161
C	Maison POS-A	205
1	Description de la maison	206

2	Modèle thermique et résultats attendus	207
2.1	Caractéristiques thermiques	207
2.2	Performances énergétiques attendues	208
D	Requêtes http pour les webserveurs	211
1	Accès à l'aide sur les différentes requêtes disponibles	211
2	Accès à la liste des bâtiments dont l'historique est disponible	212
3	Accès aux noms des services et des variables disponibles	212
E	Facteurs traités lors de l'analyse de Morris	215
F	Maison OSTWALD	221
1	Description thermique	221
2	Liste des capteurs installés	224

Remerciements



Au terme de ce long parcours que fut cette thèse, je tiens à remercier dans un premier temps tous les membres du jury. Merci à Jean-Jacques Roux et à Stéphane Lassue pour leurs rapports et leurs questionnements pointilleux et pour le moins pertinents. Merci à Laurent Mora et à Monika Woloszyn d'avoir examiné mes travaux de thèse et d'avoir assisté à la soutenance malgré la tempête.

Je souhaite ensuite adresser mes remerciements à l'équipe de choc qui m'a encadrée et qui m'ont épaulée en toute situation : un grand merci donc à Frédéric Wurtz et Etienne Wurtz, mes deux directeurs de thèse qui ont su me faire aller dans la bonne direction pour mener à bien ces travaux, merci à Philippe Maréchal pour le suivi au jour le jour et l'aide technique.

Cette thèse s'est faite entre deux laboratoires : le LGEB du CEA-INES et le G2eLab, et également, lors d'un petit détour, au Brésil.

Au LGEB, je remercie toute l'équipe, qui s'est montrée très chaleureuse, joyeuse et attentive. A ceux qui ont partagé mon bureau au cours de ces trois ans - Lisa, Clément (KIKIIII!!!), Thomas, Patricia, Léa et Grégoire : merci d'avoir supporté mes agacements et d'avoir partagé mes moments de joie. Lisa, ma grande partenaire de thèse! Merci à toi pour les fous rires, les sorties, le défoulement via le squash, et j'en passe!

A l'INES, j'ai également cotoyé de nombreuses personnes bienveillantes qui m'ont permis de passer ces quelques années de manière plus sereine : merci à la plateforme Formation et Evaluation, à Sophie, à Alain, à Xavier et Pierre que je ne remerciais jamais assez pour la tranche de vie passée ensemble, merci au LOCIE, à Nico, Yannis, Sarah et Jeanne, pour ces pauses midi passées sous le signe de la détente et de la convivialité.

Au G2elab, il y a les amis du début qui m'ont intégrée à ce grand laboratoire et ceux de la fin qui m'ont apprivoisée dans mon pire état. Tous m'ont soutenu au moment où j'en avais besoin. Merci à vous! Une mention spéciale évidemment pour Audrey et Vincent, pour les rigolades sans fin, et à Gaspard, pour ses... « blagues » mémorables, et à mes co-bureaux de fin Marie et Camille, pour les discussions et les craquages de nerfs.

xiv | Remerciements

Ao Brasil, muito obrigada a todos. Eu estou muito feliz para ter o conhecido. Eu vou sempre lembrar da Rep BMQ em Guara e de vocês : a Didi, a Ana Julia, a Nayara, a Flavia, a Paula e a Xulia. Muitos beijos e serão sempre bem vindas aqui a minha casa na França. Em Floripa, que noites com vocês! Muito obrigada o Rogério, o Thiago, o Cassio, a Sasa, a Fran, a Kauana e a Mariana! Para a Paz, minha grande amiga! Estou muito orgulhosa para ter o seu conhecido. Você é uma pessoa maravilhosa. Minha estadia em Floripa não teria sido o mesmo sem você. Você vai sempre estar no meu coração.

A mes amis. A celles que je connais depuis... très très longtemps : Pauline, Delphine et Marine. Merci d'être toujours là quand le besoin se fait sentir et de savoir toujours m'écouter et me remonter le moral. A ceux de l'école : JB, Haris, Yannis, Elisa, Alex. Merci pour les moments rares (bien qu'il faut remarquer on s'améliore d'année en année) mais toujours parfaits! Et, bien évidemment, merci à Félix pour savoir toujours me dérider dans les moments compliqués.

Finalement, un grand merci à ma famille, qui m'accompagne de tout leur coeur et sans qui cette thèse n'aurait pas été la même...

Table des figures



I.1	Répartition des émissions des GES en 2013 (ADEME [2])	6
I.2	Evolution entre 1970 et 2014 de la consommation d'énergie (ADEME [2])	6
I.3	Exigences réglementaires de consommation énergétique des bâtiments neufs (ADEME [2])	7
I.4	Répartition par secteur de la consommation d'électricité en 2015 (Cavaud et al. [25])	7
I.5	Evolution des consommations par poste dans le résidentiel (ADEME [2])	8
I.6	Capacité installée par filière - Scénario avec acceptabilité modérée des ENR (ADEME [3])	9
I.7	Représentation d'un bâtiment intelligent (Arthur [8])	12
I.8	Schéma de la procédure IPMVP (EVO [43])	17
I.9	Méthode ASHRAE adaptée de ASHRAE [9]	20
II.1	Protocole de la méthode QUB - Prandaud [77].	28
II.2	Protocole de la méthode ISABELE (Boisson et Bougié [18]).	29
II.3	Architecture de la méthodologie développée.	33
II.4	Maison POS-A - Consommations des systèmes.	37
II.5	Résultats de simulations thermiques dynamiques avec différents fichiers météo - Sidler, [94]	38
II.6	Synthèse des méthodes d'AS (coût en nombre d'évaluations du modèle vs. complexité du modèle) - Iooss [55], adapté.	42
II.7	Méthode de Morris : exemples de trajectoires, $r = 3$, $N = 3$, $Q = 3$ - Munaretto [70].	43
II.8	Représentation classique des résultats d'une analyse de Morris - Iooss [54].	44

II.9	Schéma général d'un algorithme de rejet.	47
II.10	Schéma récapitulatif de l'algorithme ABCPMC.	50
II.11	Evolution de la distribution a posteriori lors de l'application de l'algorithme ABCPMC	51
II.12	Schéma récapitulatif du fichier Resampling.py.	54
II.13	Schéma récapitulatif du fichier ScheduleMesures.py.	54
II.14	Schéma récapitulatif du fichier Class_IDF.py.	55
II.15	Schéma récapitulatif du fichier Class_Samples.py.	55
II.16	Schéma récapitulatif du fichier Morris.py.	56
II.17	Schéma récapitulatif du fichier Abcpmc.py.	57
II.18	Architecture et techniques de la méthodologie développée.	59
III.1	Maison POS-A : modèle Sketchup avec coupe sur la chambre parentale	67
III.2	Maison POS-A : modèle de la chambre parentale avec composition des parois .	67
III.3	Simulation des absences dans le logiciel EnergyPlus	67
III.4	Evolution de la distribution à posteriori lors de l'application de l'algorithme ABCPMC.	71
III.5	Résultat de l'analyse de Morris sur tous les paramètres pour l'échantillon hiver	73
III.6	Période hivernale - Température - ABCPMC : Résultat de l'algorithme pour l'épaisseur de l'isolant	74
III.7	Période hivernale - Température - ABCPMC : Résultat de l'algorithme pour la conductivité de l'isolant	74
III.8	Ecart de températures : résultat de l'analyse de Morris sur tous les paramètres pour l'échantillon été	75
III.9	Période estivale : comparaison des températures intérieures avec et sans isolation des parois extérieures.	76
III.10	Période estivale - Température - ABCPMC : Résultat de l'algorithme pour l'épaisseur de l'isolant *Faire avec 5000 valeurs	77
III.11	Ecart de consommations électriques (chauffage) : résultat de l'analyse de Morris sur tous les paramètres	79

III.12	Résultat de l'algorithme ABCPMC appliqué à l'écart de la consommation électrique de la chambre parentale	80
III.13	Résultat de l'algorithme ABCPMC appliqué à l'écart de températures pour différentes durées de simulation	82
III.14	Résultats de l'algorithme ABCPMC appliqué à l'écart de températures selon différentes durées d'absence	83
III.15	Moyennes pour chaque durée d'absence des résultats de l'algorithme ABCPMC appliqué à l'écart de températures	84
III.16	Ecart-types (bornes supérieure et inférieure) pour chaque durée d'absence des résultats de l'algorithme ABCPMC appliqué à l'écart de températures	84
III.17	Tendance globale des résultats de l'algorithme ABCPMC appliqué à l'écart de températures	85
III.18	Evolution du résultat selon le degré de défaut inséré.	87
III.19	Epaisseurs entre 10 et 13 cm : évolution du résultat selon la durée d'absence.	88
III.20	Epaisseurs entre 14 et 17 cm : évolution du résultat selon la durée d'absence.	88
III.21	Epaisseurs entre 10 et 17 cm : évolution du résultat selon la durée d'absence.	89
III.22	Epaisseurs entre 10 et 13 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.	90
III.23	Epaisseurs entre 14 et 17 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.	91
III.24	Epaisseurs entre 10 et 17 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.	91
III.25	Analyse de Morris : facteurs les plus influents lors de la recherche d'un défaut global.	93
III.26	Altération de l'isolation de la chambre parentale : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver	94
III.27	Altération de l'isolation de la chambre parentale : Résultat de l'algorithme ABCPMC pour l'épaisseur de l'isolant	95
III.28	Maison POS-A - Parois	96
III.29	Altération de l'isolation de la façade est : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver	97

III.30	Altération de l'isolation de la façade sud : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver	97
III.31	Altération de l'isolation de la façade nord : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver	98
III.32	Altération de l'isolation de la façade nord : Résultat de l'algorithme ABCPMC pour l'épaisseur de l'isolant	99
III.33	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	103
III.34	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	103
III.35	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	104
III.36	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	105
III.37	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	105
III.38	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	106
III.39	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	106
III.40	Ecart de températures pour l'échantillon hiver selon le défaut inséré.	107
III.41	Ecart de températures pour l'échantillon hiver selon le défaut inséré - Ecarts majeurs.	108
III.42	Ecart de températures pour l'échantillon hiver selon le défaut inséré - Ecarts mineurs.	108
III.43	$C_{T_{int}}$ pour l'échantillon hiver selon le défaut inséré.	109
IV.1	Maison Ostwald - Façades sud-ouest et sud-est.	117
IV.2	Maison Ostwald - Modèle thermique.	118
IV.3	Maison Ostwald - Défauts au niveau des ouvrants.	118
IV.4	Maison Ostwald - Ponts thermiques.	119
IV.5	Maison Ostwald - Défaut d'isolation des rampants.	120
IV.6	Maison Ostwald - Consommations sur le mois de mars 2017.	123
IV.7	Maison Ostwald - Scénario de puissance électrique due à l'usage selon les mesures - $P_{max} = 3.28$ W.	124
IV.8	Maison Ostwald - Température extérieure selon le fichier météorologique. . . .	125

IV.9	Maison Ostwald - Température intérieure en fonction de l'insertion des mesures dans le modèle, comparée à celle mesurée.	126
IV.10	Maison Ostwald - Température intérieure en fonction de l'insertion des mesures dans le modèle, comparée à celle mesurée - Réduit sur la période d'absence. . .	127
IV.11	Maison Ostwald - Résultat de l'analyse de Morris sur tous les paramètres. . .	129
IV.12	Maison Ostwald - Matériaux des rampants et des murs extérieurs.	130
IV.13	Maison Ostwald - Résultat de l'analyse de Morris sur tous les paramètres. . .	130
B.1	Schéma récapitulatif du fichier Resampling.py.	152
B.2	Schéma récapitulatif du fichier ScheduleMesures.py.	152
B.3	Schéma récapitulatif du fichier Morris.py.	160
B.4	Schéma récapitulatif du fichier Abcpmc.py.	161
C.1	Maison Pos-A.	206
C.2	Maison POS-A - Plan RDC.	206
C.3	Maison POS-A - Plan R+1.	206
C.4	Maison POS-A - Modèle Sketchup.	208
C.5	Maison POS-A - Modèle Sketchup : coupe chambre parents.	208
C.6	Maison POS-A - Positionnement des matériaux sur le modèle 3D.	208
D.1	Requête webservice : getAllVariableServices.	212

Liste des tableaux



II.1	Conditions pour identifier une période d'absence.	36
II.2	Maison POS-A - Périodes d'absence identifiées grâce à l'analyse des capteurs.	36
III.1	Variables utilisées et indicateurs d'écart équivalents.	68
III.2	Légende figure III.8.	76
III.3	Légende figure III.11.	80
III.4	Légende figure III.26.	94
III.5	Défauts testés.	101
IV.1	Maison Ostwald - Ecart entre la température mesurée et celle simulée après mise à jour du modèle.	127
IV.2	Maison Ostwald - Ecart entre la température mesurée et celle simulée après mise à jour du modèle - Réduit à 7°C.	128
IV.3	Légende figure IV.11.	129
IV.4	Légende figure IV.13.	131
C.1	Caractéristiques des murs extérieurs (de l'extérieur vers l'intérieur).	207
C.2	Ponts thermiques considérés	207
C.3	Bilan énergétique prévisionnel	209

Nomenclature



Acronymes

<i>A</i>	Average - Moyenne des mesures
<i>ABCPMC</i>	Algorithm Bayesian Computation - Population Monte Carlo
<i>AQC</i>	Agence Qualité Construction
<i>AS</i>	Analyse de Sensibilité
<i>BEPOS</i>	Bâtiment à Energie Positive
$C_v(RMSE)$	Coefficient of Variation of the Root Mean Square Error
<i>ECS</i>	Eau Chaude Sanitaire
<i>EF</i>	Energie Finale
<i>EP</i>	Energie Primaire
<i>GPE</i>	Garantie de Performance Energétique
<i>GPEI</i>	Garantie de Performance Energétique Intrinsèque
<i>GRE</i>	Garantie de Résultats Energétiques
<i>HLC</i>	Heat Loss Coefficient - Coefficient de pertes de chaleur
<i>IPMVP</i>	Protocole International de Plans de Mesures et Vérification
<i>ITI</i>	Isolation par l'intérieur
<i>MBE</i>	Mean Bias Error - Biais moyen
<i>MEPOS</i>	Maison à Energie Positive
<i>PEI</i>	Performance Energétique Intrinsèque
<i>R + 1</i>	Premier étage
<i>RDC</i>	Rez De Chaussée
<i>RMSE</i>	Root Mean Square Error - Erreur quadratique moyenne

STD Simulation Thermique Dynamique

Lettres grecques

ϵ_t	Critère d'acceptabilité à l'itération t pour l'algorithme ABCPMC
ϵ_c	Critère d'acceptabilité à l'itération T pour l'algorithme ABCPMC
ϵ_{max}	Critère d'acceptabilité à l'itération 1 pour l'algorithme ABCPMC
$\eta(x \theta_i^{(t)})$	Probabilité d'avoir x avec les valeurs des facteurs $\theta_i^{(t)}$ pour l'algorithme ABCPMC
λ	Conductivité des matériaux
μ	Moyenne des effets élémentaires
μ^*	Moyenne absolue des effets élémentaires
$\omega_i^{(t)}$	Poids statistique attribué aux facteurs $\theta_i^{(t)}$ pour l'algorithme ABCPMC
$\pi(\theta y)$	Inférence bayésienne lorsque dépend d'informations externes aux données y
$\pi(\theta)$	Densité de probabilité lorsque dépend d'informations externes aux données y
$\pi(\theta_i^{(t)})$	Loi a priori du facteur $\theta_i^{(t)}$ pour l'algorithme ABCPMC
σ	Ecart-type des effets élémentaires
θ	Facteur d'entrée (paramètre)
$\theta_i^{(t)}$	Facteur i à l'itération t généré à partir de la loi a posteriori trouvée à l'itération t-1 pour l'algorithme ABCPMC

Symboles

$\mathcal{N}(\theta_i^*, \sigma_i^2)$	Loi normale d'espérance θ_i^* et de variance σ_i^2
$\pm x$	Incertitude pour l'analyse de Morris
A^c	Evènement qui se réalise si et seulement si l'évènement A ne se réalise pas
C_{Chauf}	$C_v(RMSE)$ calculé entre la consommation électrique due au chauffage C_{elec} mesurée et celle simulée
C_{elec}	Consommation électrique due au chauffage
$d(x, y)$	Distance, écart entre x (simulation) et y (mesure)
E	Effet élémentaire
$f(\theta y)$	Inférence bayésienne lorsque dépend d'informations internes aux données y
$f(\theta)$	Densité de probabilité lorsque dépend d'informations internes aux données y

M	Données mesurées
N_S	Nombre de simulations
$P(A E)$	Probabilité de l'évènement A sachant l'évènement E
$P(A)$	Probabilité de l'évènement A
$P(B E)$	Probabilité de l'évènement B sachant l'évènement E
$P(B)$	Probabilité de l'évènement B
$P(E A)$	Probabilité de l'évènement E sachant l'évènement A
$P(E)$	Probabilité de l'évènement E
S	Données simulées
T	Nombre total d'itérations pour l'algorithme ABCPMC
t	Itération variant de 1 à T pour l'algorithme ABCPMC
T_{int}	Température intérieure
x	Variable simulée pour l'algorithme ABCPMC
y	Variable mesurée pour l'algorithme ABCPMC

Introduction



En 2014, le bâtiment représente 45% des consommations énergétiques nationales et est responsable de 20% des émissions de gaz à effet de serre (GES). Les gouvernements ont alors mis en place différentes politiques, à l'échelle nationale et internationale, visant des réductions de consommations et des émissions des GES du secteur du bâtiment. Nous avons donc assisté à la naissance des bâtiments performants, des bâtiments à énergie positive puis des bâtiments intelligents et connectés. Nous avons donc eu d'abord des améliorations sur l'enveloppe du bâtiment, permettant des déperditions thermiques moindres, et sur les systèmes du bâtiment (chauffage, refroidissement et climatisation, ventilation, ECS, éclairage) afin qu'ils consomment moins. Ensuite, nous avons voulu avoir des bâtiments qui produisent leur énergie. Puis, nous voulions optimiser la consommation et la production d'énergie. Il a donc fallu rendre le bâtiment intelligent qui permet, selon des conditions définies, de choisir le meilleur moment d'utiliser la production sur site ou bien le réseau, ou encore, lorsqu'il s'agit seulement d'un bâtiment performant, quand il est le plus idoine d'utiliser les gros appareils électroménagers. Cependant, ces bâtiments intelligents fonctionnant selon des algorithmes manquaient d'interactions avec les usagers qui pouvaient se sentir lésés, notamment en termes de confort, et souhaitaient alors reprendre le contrôle du bâtiment en ajoutant des radiateurs électriques s'ils ont trop froid, ou bien en ouvrant les fenêtres en été car le courant d'air extérieur les rafraichit à l'instant t. Il y a donc eu apparition des technologies permettant à l'occupant de gérer son bâtiment, et de décider, en fonction de son emploi du temps, comment le bâtiment doit se comporter.

Les bâtiments intelligents et les bâtiments connectés génèrent ainsi un grand nombre de données. De ce constat, nous nous sommes interrogés sur comment utiliser toutes ces données.

En parallèle, la communauté scientifique s'attèle au concept de garantie de performance énergétique (GPE). En effet, sur le papier, nous construisons des bâtiments de plus en plus performants, or, les retours d'expérience ont montré que les performances réelles n'étaient pas aussi bonnes que celles prévues, la faute soit à l'occupant, qui a des conditions de confort supérieures à celles prises en compte lors des simulations, mais également à la construction : des

2 | INTRODUCTION

matériaux différents ont été utilisés, ou mal posés, les systèmes ont des dysfonctionnements. Pour identifier ces derniers défauts, différents tests ont été mis au point : QUB, ISABELE, Co-heating, etc. Ceux-ci ont tous un point commun : ils nécessitent un certain temps de mesures sur site, souvent plusieurs jours voire une semaine, avec une équipe technique, et des occupants hors site.

Devant ce constat, nous nous sommes demandés s’il était possible d’utiliser les données envoyées par le bâtiment connecté pour détecter ces fameux défauts de construction et c’est la problématique à laquelle nous avons essayé de répondre par ces travaux de thèse.

Ce manuscrit se présente sous la forme de quatre chapitres, suivis d’une conclusion générale et les perspectives à ces travaux. Dans le premier chapitre, nous définissons notre environnement de travail, en mettant en relief le contexte énergétique ainsi qu’en définissant les différents termes que nous allons utiliser tout au long de ce manuscrit. Dans le deuxième chapitre, nous présentons la méthodologie que nous avons mise en place et nous effectuons ensuite un état de l’art des différentes techniques que nous pourrions utiliser pour finalement présenter celles que nous avons choisies. Dans le troisième chapitre, nous éprouvons notre méthodologie, par la simulation et la création de défauts « virtuels », sous différents critères afin d’en évaluer les performances mais également les limites. Pour cela, nous allons modéliser des défauts et nous essaierons de les retrouver par la simulation. Dans le quatrième et dernier chapitre, nous appliquerons notre méthodologie à un cas réel, issu du projet COMEPOS, en comparant cette fois les mesures qui sont effectuées en continu sur site grâce à différents capteurs et le modèle qui a permis initialement d’estimer les performances de cette maison. Nous concluerons ensuite sur ces travaux, entre ce qui a été fait et démontré et ce qu’il reste à effectuer par la suite.

Chapitre I

Vers la garantie de performances

Table des matières

1 Un contexte énergétique particulier : une prise de conscience mais des consommations toujours élevées.	6
1.1 Le rôle du bâtiment pour atteindre les objectifs environnementaux	6
1.2 Le bâtiment et l'électricité : un consommateur producteur	7
2 Les enjeux du bâtiment connecté	9
2.1 Présentation et définition d'un bâtiment connecté pour ces travaux de thèse	9
2.2 Les développements académiques	11
2.3 Le marché des offres grands publics	11
2.4 Le bâtiment intelligent : un atout pour la transition énergétique	12
3 La ou plutôt les garanties de performances	13
3.1 Définition de la garantie de performances (GP)	13
3.1.1 Garantie de Performance Energétique Intrinsèque (GPEI)	14
3.1.2 Garantie de Résultats Energétiques (GRE) sur l'usage	15
3.2 Plans mis en oeuvre pour y répondre	16
3.2.1 Contrat de Performance Energétique (CPE)	16
3.2.2 International Performance Measurement & Verification Protocol (IPMVP)	17
3.2.3 ASHRAE : Guideline 14-2002	19
4 Besoins ressentis et notre proposition pour y répondre	20

Résumé



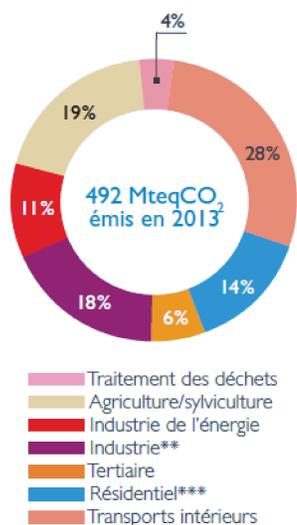
Nous proposons dans ce chapitre une mise en contexte de la problématique traitée dans ce manuscrit de thèse. Nous abordons dans un premier temps les enjeux énergétiques et environnementaux liés au secteur du bâtiment et notamment son rôle majeur dans la consommation d'électricité en France. Nous définissons dans une deuxième partie le bâtiment connecté et nous explorons ses différents usages et les développements gravitant autour. Dans la troisième partie, nous exposons la définition ou plutôt les définitions de la garantie de performances (GP) à travers les concepts de la GPEI, garantie de performances énergétiques intrinsèques, et de la GRE, garantie de résultats énergétiques. Nous parlons ensuite des moyens mis en oeuvre pour répondre à ces concepts avec notamment le protocole IPMVP définissant des structures de mesure et de vérification des performances, appelées plans M&V. Nous concluons ce chapitre en exprimant les besoins ressentis au niveau de la GP et du bâtiment connecté puis en proposant une nouvelle approche pour répondre à ces besoins.

1 Un contexte énergétique particulier : une prise de conscience mais des consommations toujours élevées



1.1 Le rôle du bâtiment pour atteindre les objectifs environnementaux

Depuis quelques années, nous nous trouvons dans un contexte énergétique spécifique où les différents pays ont pris conscience de leur consommation énergétique et de leur impact sur notre environnement. Le premier accord majeur en ce sens passé entre les Nations Unies fut l'accord de Kyoto [71] lors de la COP 3 en 1998 (version définitive). En 2015, Paris a accueilli la COP 21 dont le but était de convenir d'un accord permettant de contenir le réchauffement climatique à 2°C. Le bâtiment est perçu comme un point clé, stratégique, pour la lutte contre le réchauffement climatique, et pour cause : dans son rapport « Climat, Air et Energie », l'ADEME [2] montre qu'il est responsable en France en 2013 de 20% des émissions des gaz à effet de serre (GES), source en grande partie de ce réchauffement (figure I.1). Le secteur du résidentiel est à lui seul responsable de 14% des émissions de GES et de 45% des consommations énergétiques nationales (figure I.2).



* Hors UTGF (Utilisation des Terres, leur Changement d'affectation et la Forêt)
 ** Procédés utilisant des solvants inclus
 ***Gaz fluorés et solvants compris

Figure I.1 – Répartition des émissions des GES en 2013 (ADEME [2])

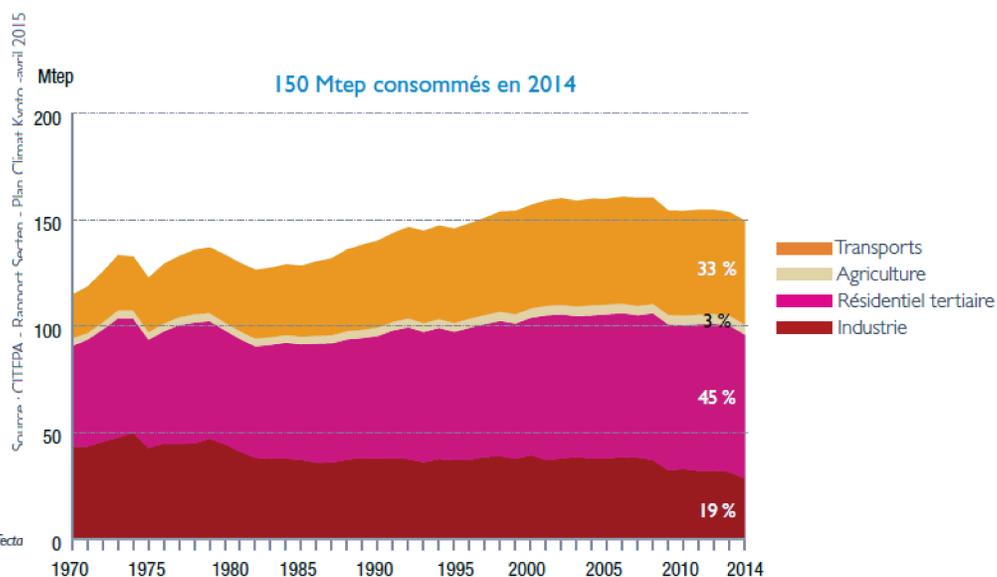


Figure I.2 – Evolution entre 1970 et 2014 de la consommation d'énergie (ADEME [2])

Le gouvernement a donc renforcé la réglementation thermique (RT) au fil des ans. La figure I.3 présente cette évolution des exigences énergétiques pour la construction de nouveaux édifices depuis 2000. On peut constater que la RT est devenue de plus en plus restrictive. En

1. UN CONTEXTE ÉNERGÉTIQUE PARTICULIER : UNE PRISE DE CONSCIENCE MAIS DES CONSOMMATIONS TOUJOURS ÉLEVÉES | 7

2000, il était en effet demandé aux constructeurs des bâtiments consommant moins de 190 kWh_{EP}/m²/an¹. En 2012, l'exigence est descendue à 50 kWh_{EP}/m²/an pour cinq sources, ou postes, de consommation : le chauffage, le rafraîchissement, l'eau chaude sanitaire (ECS), la ventilation et l'éclairage. Pour 2020, il a été annoncé que les nouveaux bâtiments construits soient à énergie positive, c'est-à-dire qu'ils produiraient plus d'énergie qu'ils n'en consomment.

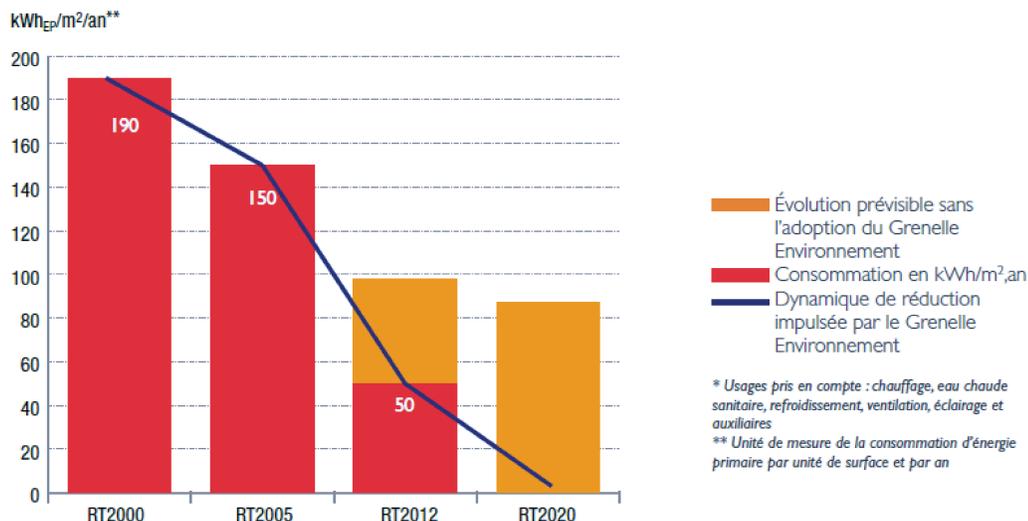
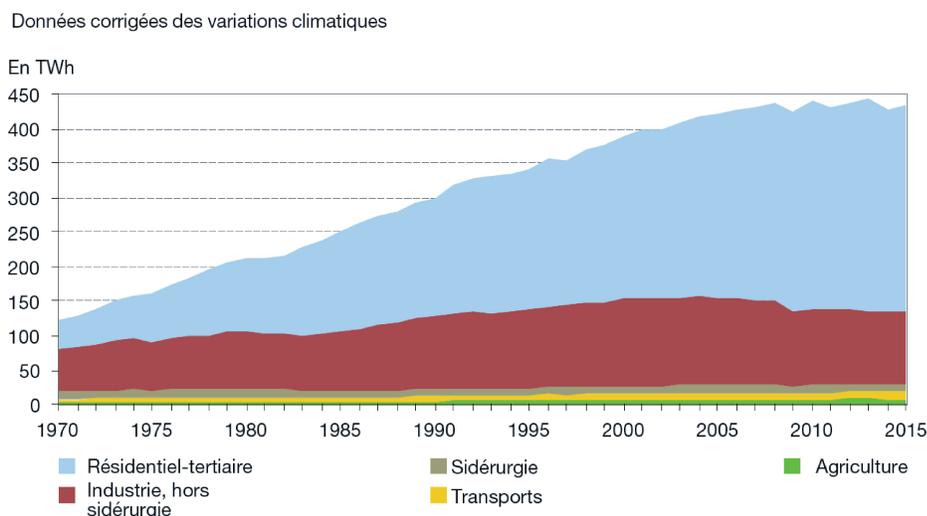


Figure I.3 – Exigences réglementaires de consommation énergétique des bâtiments neufs (ADEME [2])

1.2 Le bâtiment et l'électricité : un consommateur producteur



Sources : calculs SOeS, d'après l'enquête sur le transport et la distribution d'électricité, RTE, Enedis et Rica

Figure I.4 – Répartition par secteur de la consommation d'électricité en 2015 (Cavaud et al. [25])

1. EP : Energie Primaire

Si l'on parle du bâtiment, il est essentiel de parler d'électricité. En effet, comme le montre la figure I.4, le bâtiment était, en 2015, le premier consommateur avec deux tiers de la consommation finale d'électricité.

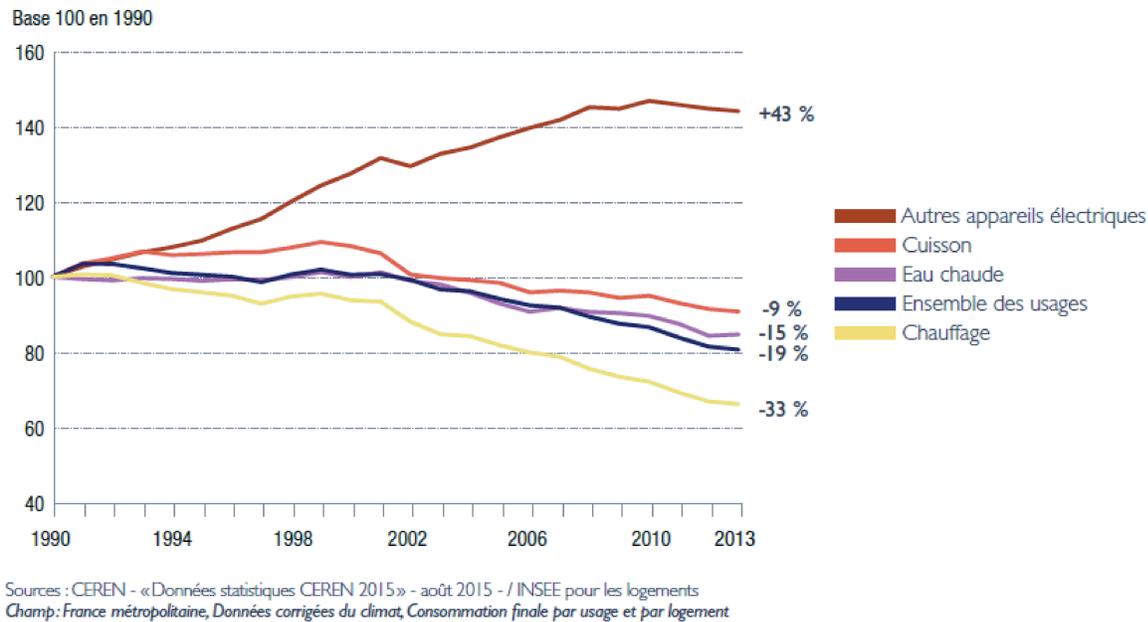


Figure I.5 – Evolution des consommations par poste dans le résidentiel (ADEME [2])

La figure I.5 présente l'évolution des consommations énergétiques par poste entre 1990 et 2013. Celles des postes considérés dans la RT ont diminué : plus les exigences en matière de réduction des consommations et de performance énergétique ont augmenté, plus les consommations sur les postes visés ont diminué. On peut cependant remarquer que la consommation des appareils électriques a largement augmenté depuis 1990, malgré une légère décroissance entre 2010 et 2013.

Les appareils électriques sont directement liés à l'occupant. Il est donc de plus en plus primordial de prendre en compte son comportement et l'utilisation qu'il fait des appareils électriques lors de la prévision des consommations d'un bâtiment. Il s'agit d'un point étudié lors de nombreux travaux de recherche, que nous précisons par la suite (II.2). Même si son comportement est pris en compte, il est difficile de changer ses habitudes. Il est donc essentiel d'étudier la possibilité de contrebalancer ces consommations importantes d'électricité par la production d'électricité, et donc d'évaluer le potentiel du bâtiment dans ce domaine.

Dans son rapport « Vers un mix énergétique 100% renouvelable », l'ADEME [3] étudie quatre scénarios : un de référence, un prenant en compte des avancées technologiques évolutionnaires, un étudiant le cas d'un renforcement difficile du réseau et un avec une acceptabilité modérée (les gisements utilisant de la surface au sol ont été particulièrement contraints). Si l'on observe les résultats de ce dernier cas, la production PV en toiture possède la plus grande

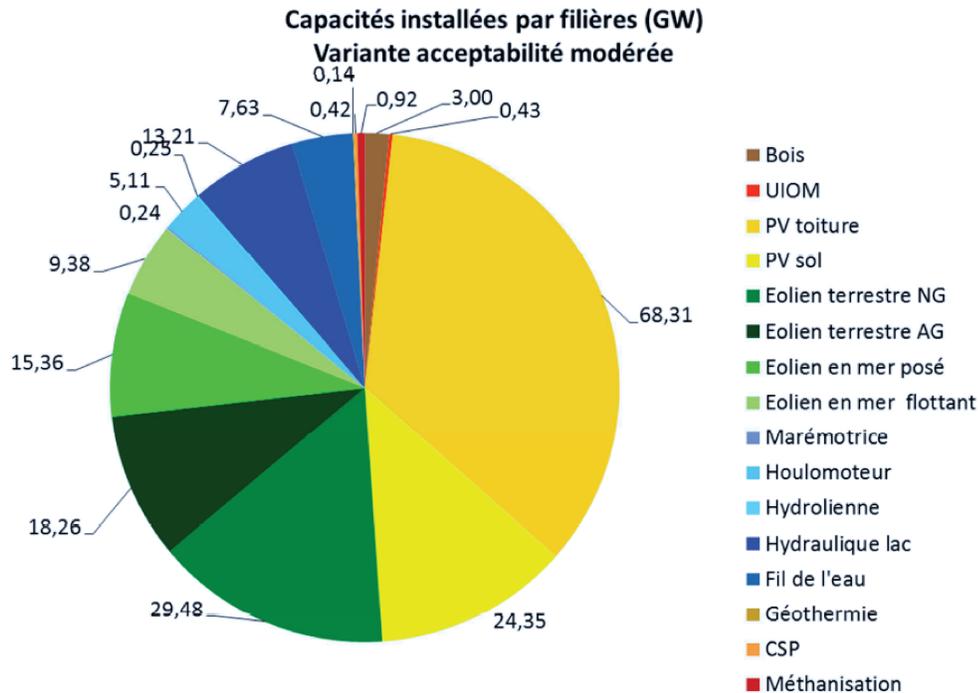


Figure I.6 – Capacité installée par filière - Scénario avec acceptabilité modérée des ENR (ADEME [3])

capacité annuelle avec 68 GW (figure I.6) ce qui correspond à une production annuelle de 88 TWh. Le bâtiment est un grand consommateur d'électricité, de plus en plus par les usages des occupants, mais également un potentiel producteur majeur de cette énergie.

2 Les enjeux du bâtiment connecté △

2.1 Présentation et définition d'un bâtiment connecté pour ces travaux de thèse

Un bâtiment connecté peut être défini comme un bâtiment actif et communicant. Il contient des capteurs et parfois des actionneurs qui permettent respectivement de récupérer des mesures et d'agir sur les systèmes afin de répondre aux demandes de l'occupant. Par exemple, ce dernier va pouvoir modifier la température de son habitat grâce à son smartphone en fonction des températures de confort de l'occupant et/ou du coût de l'énergie ou encore fermer les volets en été en fonction de l'ensoleillement afin de préserver une température relativement fraîche dans une pièce, toujours selon les informations entrées par l'occupant. Dans la plupart des cas de bâtiments connectés, les données enregistrées grâce aux capteurs sont remontées sur des serveurs externes et sont ainsi disponibles à distance.

La notion de bâtiment connecté est souvent proche de celle de bâtiment intelligent qui va pouvoir, par exemple, optimiser l'usage des différents appareils domestiques afin de décharger le réseau et ainsi de limiter les pics de demande en électricité. Buckman et al. [23] présentent la différence entre les termes « intelligent building » et « smart building ». Le premier a la notion de réactivité : il réagit selon les exigences des occupants, de l'entreprise ou de la société de gestion énergétique. Le second a la notion d'adaptabilité : avec les informations qu'il détient, il doit être capable de s'adapter aux changements et à l'environnement sur une certaine période de temps.

En français, la différence de vocabulaire ne se ressent pas car les deux se traduisent par « bâtiment intelligent ». Il faut donc converger vers une définition claire lorsque l'on parle de bâtiment intelligent dans un groupe de travail. Dans le cas de ces travaux de thèse, le terme sera employé au plus simple degré, c'est-à-dire avec la définition d'un « intelligent building », et ce car seulement le fait que les mesures soient disponibles à distance nous intéresse, et non pas forcément une adaptabilité. Un bâtiment intelligent est donc, au sens le plus basique, avant tout un bâtiment connecté. Ainsi ces deux termes, « bâtiment connecté » et « bâtiment intelligent », seront employés avec la même définition dans ce manuscrit.

Celui-ci existe grâce au développement de l'internet des objets (IoT : « Internet of Things »). Rose et al. [86] présente une vue d'ensemble de l'IoT. Ce terme a été pour la première fois utilisé par Ashton [11] en 1999 et le définit comme un système où des objets physiques transmettent des informations à l'Internet. Il est arrivé à cette définition en mettant en avant le fait qu'Internet sait seulement gérer des informations mais que nous, Hommes, sommes entourés d'objets, sur lesquels nous avons fondé notre société, notre économie, etc. De ce fait, Ashton imagine que, si les ordinateurs pouvaient « comprendre » nos objets, il serait possible de tout compter, de tout suivre, et donc de diagnostiquer les défaillances des objets nous entourant.

Dans la première partie de ce chapitre 1, nous avons vu que les futurs bâtiments seront probablement à énergie positive, et donc générateurs d'électricité, via des panneaux solaires photovoltaïques, des éoliennes, etc.), ainsi que stockeurs grâce à la mise en place de batteries. Dans ce contexte-ci, le bâtiment connecté va devenir un atout non négligeable.

Grâce à lui et à des algorithmes capables d'analyser et de manipuler toutes les données que ce type de bâtiments peut générer, tels que ceux utilisés pour le Big Data, il est désormais possible de prévoir les consommations et la production d'électricité et ainsi d'optimiser son utilisation en fonction de sa provenance (réseau, production locale directe ou stockée). Nous proposons dans cette deuxième partie de balayer l'éventail des développements du bâtiment connecté, permettant la gestion énergétique de celui-ci.

2.2 Les développements académiques

Le bâtiment connecté a été le sujet de nombreuses recherches et expérimentations dans le monde académique. Zhao et al. [107] montrent que l'IoT contribue à l'efficacité énergétique dans les bâtiments. Si l'on se focalise sur la France, plusieurs projets visent le développement du bâtiment connecté. Ces travaux de thèse s'effectuent dans le cadre de l'un d'entre eux, le projet COMEPOS visant le développement du concept de maison à énergie positive et connectée. Dans ce projet, on cherche à valoriser le bâtiment connecté pour le bon fonctionnement des maisons et des systèmes innovants mis en oeuvre dans le cadre de la production d'énergie. Un autre projet, le projet ANR INVOLVED [6] vise quant à lui le développement d'un e-consultant pour l'efficacité énergétique. Il s'agit ici d'amener l'occupant à se sentir concerné par son bâtiment et de lui fournir simplement des conseils faciles à mettre en oeuvre.

2.3 Le marché des offres grands publics

Le monde de la recherche n'est pas le seul à s'être emparé du domaine du bâtiment connecté : les industriels également. Deux catégories de systèmes se distinguent : d'un côté il y a ceux permettant d'être informé des variables de consommation et de confort (température, humidité et CO₂ par exemple) et de l'autre ceux permettant d'informer mais également d'agir sur l'habitat. Dans la première catégorie, on retrouve des spécialistes des technologies telles que Apple [60] et Google [61] qui permettent la gestion des différents objets connectés. Des entreprises se sont également spécialisées dans le domaine et proposent des offres complètes intégrant 3 axes principaux : la sécurité, le confort (température et qualité de l'air) et la météo. On peut citer les entreprises Netatmo [72] et Delta Dore [35]. Celles-ci ont d'abord eu ce rôle d'informateur mais elles développent également l'aspect acteur notamment sur le point de la sécurité, avec des fonctions permettant à l'occupant de fermer ses volets, d'activer l'alarme ou d'éteindre l'éclairage à distance.

Dans la seconde catégorie, on retrouve les gestionnaires énergétiques tels que Gapéo [47], Ergelis [40] et Schneider Electric, qui a développé avec Microsoft une plateforme appelée EcoStruxure [91], qui visent plutôt le secteur tertiaire. D'autres se concentrent sur le résidentiel comme par exemple Nexity qui a développé en partenariat avec Fabernovel et Applidium l'application Alfred [45]. Des constructeurs commencent à développer leur propre système, tel Flexem de Bouygues [22].

On a donc des entreprises qui visent à informer l'habitant sur son environnement (confort, sécurité, météo) et d'autres informant et agissant sur les systèmes afin de correspondre au mieux aux demandes des occupants.

2.4 Le bâtiment intelligent : un atout pour la transition énergétique

Le bâtiment intelligent peut être un atout majeur pour la transition énergétique. Dans la partie 1 de ce chapitre, il a été montré que le bâtiment est un consommateur et un potentiel producteur d'énergie majeur. Cela engendre des flux d'énergie entrant et sortant du bâtiment (représentés sur la figure I.7) qui doivent être gérés, d'où l'intérêt du bâtiment intelligent.

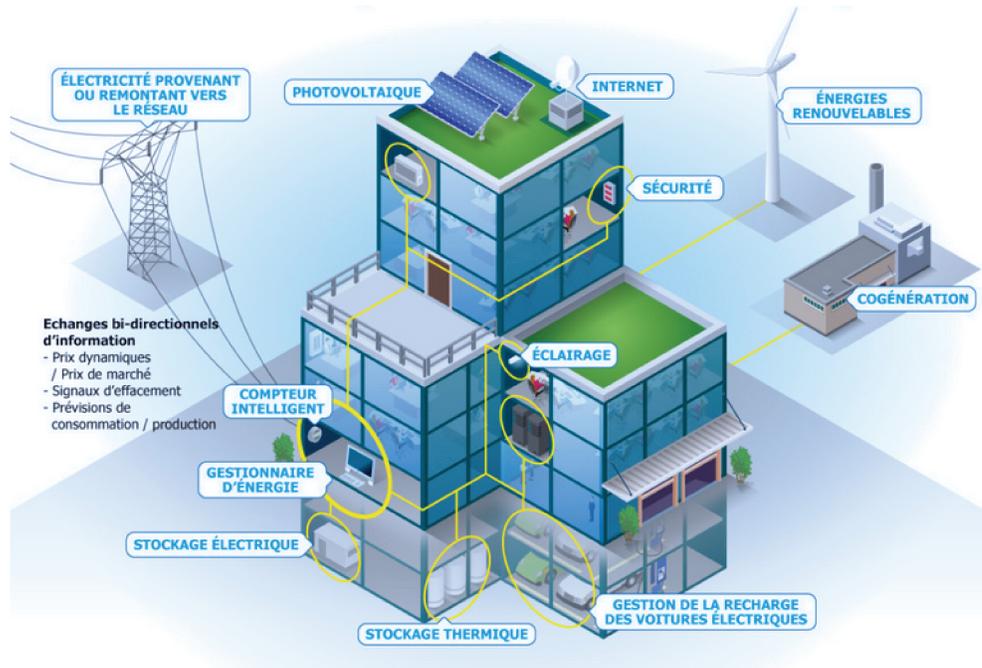


Figure I.7 – Représentation d'un bâtiment intelligent (Arthur [8])

Il permet de gérer ces flux selon différents critères. Si l'on prend l'exemple du bâtiment présenté en figure I.7, il possède des batteries (stockage électrique) et du PV en toiture. Il est également relié au réseau. Le bâtiment intelligent va dans ce cas être capable de décider quelle source d'électricité choisir en fonction de la production et de la demande sur site, mais aussi si le réseau est largement sollicité ou pas.

L'électricité produite peut aussi être utilisée en dehors du bâtiment si la production couvre plus que ses besoins et que les batteries sont pleines, comme cela peut être le cas en été. Il y a alors plusieurs possibilités. Les véhicules électriques se développant de plus en plus, l'excédent d'électricité peut servir à recharger les batteries des véhicules. Dargahi [32] explore dans sa thèse cette possibilité et notamment le cas d'une maison individuelle. On parle alors du concept « Vehicle To Home » (V2H), où l'électricité peut aller dans les deux sens : soit la batterie du véhicule est rechargée par l'excédent produit par la maison, soit la batterie est sollicitée lors de pénuries d'électricité produite sur site. L'électricité peut aussi être utilisée hors-site : elle peut être rejetée sur le réseau ou stockée à l'échelle nationale grâce à, par exemple, des systèmes « power to gas ». Le stockage de l'énergie est nécessaire pour

répondre à la problématique de l'intermittence saisonnière, qui est la principale critique faite aux énergies renouvelables. Le bâtiment intelligent permet donc une gestion optimale de ces flux énergétiques.

Le bâtiment intelligent est donc un « objet » pouvant à la fois consommer, produire et gérer son confort, sa production d'énergie, sa qualité d'air intérieur, etc. Pourtant les occupants peuvent éprouver des difficultés à comprendre le fonctionnement du bâtiment et ainsi altérer ses performances voire à rejeter ce principe du bâtiment contrôlant et agissant par lui-même. Il est alors primordial de prendre en compte ce facteur d'acceptabilité des systèmes, des capteurs, si l'on souhaite que le bâtiment intelligent soit exploité à son plein potentiel. Zelem et al. [108] mettent en avant ce point. Ils relèvent que les équipements intelligents et donc les bâtiments intelligents reposent sur l'hypothèse que les occupants coopèrent et adoptent des comportements attendus par les équipements et qu'ils deviennent ainsi des occupants « intelligents ».

3 La ou plutôt les garanties de performances

3.1 Définition de la garantie de performances (GP)

Selon le CNRTL², la garantie est définie de manière générale comme « l'action ou le moyen de garantir quelque chose à quelqu'un, le résultat de cette action impliquant la responsabilité qui en découle ». Lorsque l'on vend un bien, la garantie devient synonyme d'obligation pour le vendeur de répondre de la qualité du bien vendu. En ce qui concerne la performance, celle-ci est définie comme « le rendement, ou fiabilité, très élevé d'une machine, d'un objet, d'un matériau ».

Si l'on applique ces définitions au domaine du bâtiment, les constructeurs vendent un bien, un logement, avec certaines performances énergétiques pour un confort donné. Il est en effet important de souligner que la consommation énergétique ne s'entend qu'en fonction du confort qu'elle permet d'assurer, car, sans prise en compte du confort, le bâtiment le plus « performant » est celui qui ne consomme rien (pas de chauffage, pas de climatisation, pas de ventilation, etc.).

Cela signifie donc qu'ils ont la responsabilité de s'assurer que le logement (bâti et systèmes) fonctionne selon le rendement qu'ils ont annoncé.

Le Plan Bâtiment Durable présente dans son rapport rédigé par Huet et al. [53] plusieurs propositions pour améliorer la RT2012, avec notamment l'utilisation de « l'article L.111-

2. Centre National des Ressources Textuelles et Lexicales (<http://www.cnrtl.fr/>)

22 du Code de la Construction et de l'Habitation », définissant un seuil de consommation électrique et de température au delà duquel le bien devient respectivement « impropre à la consommation et à sa destination ». Cela permet d'avoir des valeurs concrètes sur lesquelles s'appuyer. Ainsi, ce n'est plus le juge qui décide si la performance énergétique est garantie ou non.

Selon Huet et al., la garantie de performance énergétique (GPE) a pour but de garantir une efficacité énergétique, dont la mise en oeuvre se traduit par l'obligation d'un prestataire d'atteindre les objectifs d'efficacité énergétique fixés. Deux types de garantie sont également déterminés :

- une GPE au niveau de la conception et des travaux garantissant les performances intrinsèques du bâtiment, appelée Garantie de Performance Énergétique Intrinsèque (GPEI).
- une GPE au niveau de l'exploitation et de l'usage garantissant des résultats énergétiques, appelée Garantie de Résultats Énergétiques sur l'usage (GRE).

3.1.1 Garantie de Performance Énergétique Intrinsèque (GPEI)

La GPEI intervient lorsque les travaux prévus visent des performances supérieures à celles de la RT, sous-entendant que, dans un tel cas, c'est la loi qui sert de « garantie ». Elle concerne les différentes opérations au stade de la conception et de la réalisation des travaux. Ses plus-values par rapport à la RT2012 sont principalement qu'elle aspire à des consommations énergétiques moins élevées ainsi qu'une affirmation réelle de la garantie. Alors que la RT s'adresse principalement au secteur du bâtiment neuf, la GPEI s'applique également à la rénovation, lorsque les travaux effectués ont pour objectif certains seuils de performance.

D'un point de vue juridique, la GPEI peut être définie comme un engagement contractuel entre :

- un maître d'ouvrage, ou un propriétaire, appelé « le bénéficiaire »
- un « prestataire » (maître d'œuvre, entreprises), appelé « le garant ».

On a donc un « prestataire » qui s'engage sur un niveau de consommations énergétiques dans le cas du respect d'un scénario d'utilisation et de confort précis.

Si l'on considère son périmètre d'application, il s'agit de celui de la RT2012 et de ses 5 usages (chauffage, refroidissement, ECS, ventilation, éclairage). Les consommations sont estimées à la fois en énergie primaire mais également en énergie finale, ce qui facilite la communication avec les futurs utilisateurs et qui permet le lien avec la GRE.

Du point de vue pratique, on évalue les performances à partir d'un référentiel ce qui entraîne une certification de l'ouvrage accompagnée d'autocontrôles durant les phases d'études, de chantiers et d'avant réception avec des mesures sur site.

La GPEI permet d'éviter certaines des causes d'écart entre consommations réelles et théoriques, comme les choix de conception inadaptés, des matériaux et des équipements moins performants que prévu, une mauvaise mise en œuvre, etc.

Malgré les bonnes intentions de la GPEI, il faut être attentif à différents points, pouvant être des inconvénients majeurs pour sa mise en œuvre. Elle prévoit en effet des contrôles mais ceux-ci s'accompagnent forcément d'un surcoût, car ils nécessitent un recours à des experts, des déplacements ainsi qu'un traitement des mesures, et le désagrément causé aux occupants par ces contrôles. Il faut également noter qu'elle ne prend pas en compte la dégradation des matériaux au fil du temps ni les conséquences d'un suivi et d'un entretien insuffisant ce qui peut être problématique si l'on souhaite garantir les performances sur une certaine durée.

3.1.2 Garantie de Résultats Energétiques (GRE) sur l'usage

La GRE, définie dans le rapport Jouvent Costa [29] est également un engagement contractuel entre un maître d'ouvrage, ou un propriétaire (ou locataire), appelé « le bénéficiaire » et un prestataire spécialisé, appelé « le garant », par lequel ce dernier s'engage, sur une durée spécifiée, dans le cas de travaux (neufs ou existants) ou de contrats de performance énergétique (CPE) sur des points particuliers, tels que le niveau maximal de consommations énergétiques ou un pourcentage de réduction des consommations, tout en respectant des paramètres de confort (température, qualité de l'air, etc.).

Son périmètre d'application peut être défini au cas par cas selon les besoins. Il peut aller de la garantie sur l'énergie de chauffage jusqu'à la garantie de tous les usages et de la production d'énergie sur site. Ainsi, contrairement à la GPEI, la GRE n'est pas restreinte aux cinq usages de la RT2012.

Il est primordial que les consommations soient mesurées selon une méthode simple puis corrigées selon les différents facteurs d'influence (météo, usages différents de ceux pris en compte lors de l'estimation des consommations, etc.). Ensuite, si les consommations mesurées sont supérieures à celles garanties, le « garant » subit alors une pénalité proportionnelle à la consommation excessive observée (consommation \times prix unitaire de l'énergie).

L'avantage principal de la GRE se situe dans son obligation d'un suivi précis et régulier des différentes consommations grâce à un système de suivi, par exemple un de ceux présentés à la partie 1.2.3, en adéquation avec les besoins des occupants. La GRE permet enfin d'élargir la démarche. Elle prend comme base l'énergie, de manière classique, mais peut être

étendue à l'eau, aux déchets, à la qualité de l'air, etc. Si elle prend en compte ces aspects environnementaux, elle deviendrait alors une Garantie de Résultats Environnementaux.

Comme pour la GPEI, il faut être vigilant quant au surcoût induit par la mise en place de la GRE causé par le recours à des experts, la gestion des appareils de mesure, etc. Il faut également prêter attention aux termes du contrat. La GRE s'applique seulement au périmètre défini et elle indemnise les dysfonctionnements mais elle ne contribue pas à la réparation de ceux-ci.

3.2 Plans mis en oeuvre pour y répondre

Entre les exigences de la RT et le développement du concept de garantie de performance, de nouvelles méthodologies ont été développées afin de poser des bases et des directions pour la bonne construction et l'évaluation des performances du bâtiment. Nous expliquons ici les principes du contrat de performance énergétique (3.2.1), de l'IPMVP (3.2.2) et de la Guideline 14 de l'ASHRAE (3.2.3), visant tous la garantie de performances via leur évaluation.

3.2.1 Contrat de Performance Energétique (CPE)

Ortega [74] présente en 2011 pour la ministre de l'écologie, du développement durable, des transports et du logement, un état des lieux des contrats de performances énergétiques.

Le CPE est défini, par la directive 2012/27/UE [80], comme « un accord contractuel entre le bénéficiaire et le fournisseur d'une mesure visant à améliorer l'efficacité énergétique, vérifiée et surveillée pendant toute la durée du contrat, aux termes duquel les investissements (travaux, fournitures ou services) dans cette mesure sont rémunérés en fonction d'un niveau d'amélioration de l'efficacité énergétique qui est contractuellement défini ou d'un autre critère de performance énergétique convenu, tel que des économies financières ». Grâce au CPE, le fournisseur doit indemniser le maître d'ouvrage de la totalité de l'écart observé entre la performance fixée dans le contrat et celle constatée sur site.

Sur le terrain, Ortega [74] observe trois types de contrats : (i) les CPE « Fournitures et Services », axés sur les systèmes de gestion énergétique ainsi que les équipements de production et de consommation d'énergie, (ii) les CPE « Travaux et services », axé sur l'enveloppe du bâtiment, et (iii) les CPE « Globaux », qui englobent l'enveloppe et les systèmes du bâtiment. Peu importe son type, tout CPE doit contenir quatre éléments :

- « l'objet » : il définit la quantité d'économies d'énergie prévue et non pas les travaux à mettre en oeuvre.
- « l'investissement » : il représente l'investissement financier permettant la réalisation des travaux pour l'amélioration énergétique du bâtiment concerné.

- « la garantie de performance énergétique » : elle est obligatoire et permet au maître d'ouvrage le respect des objectifs fixés en termes d'économies d'énergies.
- « la mesure des performances énergétiques » : elle est à effectuer aux différents stades des travaux : avant, pendant et après. La méthode utilisée doit être validée par le maître d'ouvrage et la société d'efficacité énergétique.

Il reste ensuite à définir une méthode pour mesurer les économies d'énergie : c'est l'objectif du protocole IPMVP décrit dans la partie suivante.

3.2.2 International Performance Measurement & Verification Protocol (IPMVP)

Le protocole IPMVP (en français [43], en anglais [44]) est comme son nom l'indique un protocole permettant d'établir un plan de Mesures et Vérifications (M&V). Un plan de M&V comprend cinq actions principales : (i) l'installation, la calibration et la maintenance des compteurs, (ii) la collecte et le traitement de données, (iii) le développement d'une méthode de calcul et d'estimations acceptables, (iv) la réalisation des calculs à partir des données mesurées et (v) le suivi, l'assurance de la qualité et la vérification des rapports par un tiers. Il vise ainsi à accroître les économies d'énergie tout en aidant le public à comprendre la gestion de l'énergie.

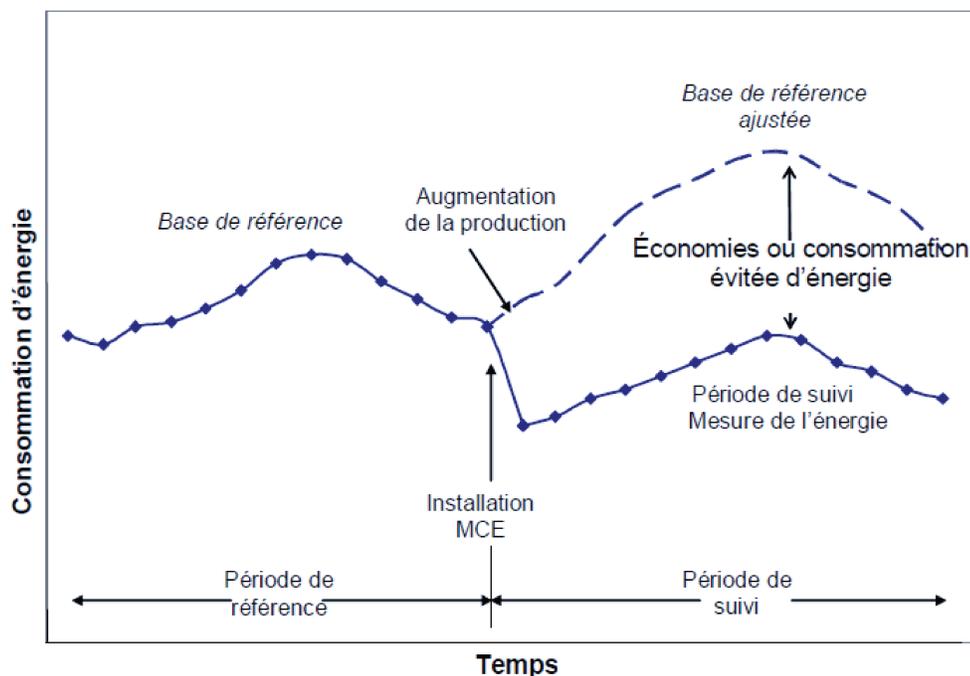


Figure I.8 – Schéma de la procédure IPMVP (EVO [43])

Les principes du protocole se basent sur la figure I.8 où sont exprimés les principaux termes employés par IPMVP. Le point central est la Mesure de Conservation de l'Énergie (MCE), correspondant à l'instant où les actions permettant les économies d'énergie sont mises

en place. IPMVP propose de définir les économies énergétiques réalisées comme indiqué par l'équation I.1.

$$\begin{aligned} \text{Economies} &= \text{Consommation (ou appel de puissance) de la période de référence} \\ &\quad - \text{Consommation (ou appel de puissance) de la période de suivi} \pm \text{Ajustements} \end{aligned} \quad (\text{I.1})$$

Il propose quatre options (A, B, C et D) selon la situation devant être évaluée, et ainsi, quatre équations permettant d'évaluer les économies d'énergie réalisées. Les options A et B se basent sur l'isolement des MCE permettant ainsi de diminuer le périmètre de mesure. L'option A se concentre sur l'isolement des paramètres clés tandis que la B étudie tous les paramètres. Ainsi, pour l'option A, l'équation I.1 peut intégrer à la fois des mesures et des estimations de paramètres. I.1 devient alors :

$$\begin{aligned} \text{Economies}_{\text{optionA}} &= \text{Valeur estimée} \\ &\quad \times (\text{Paramètre mesuré pendant la période de référence} \\ &\quad - \text{Paramètre mesuré pendant la période de suivi}) \end{aligned} \quad (\text{I.2})$$

Contrairement à l'option A, l'option B mesure toutes les quantités d'énergie de l'équation I.1. Elle aura donc tendance à donner des résultats plus précis que l'option A malgré un coût pouvant être supérieur. Ici, tous les paramètres sont mesurés, il n'y a donc pas besoin d'ajustements. I.1 devient alors :

$$\text{Economies}_{\text{optionB}} = \text{Energie de la période de référence} - \text{Energie de période de suivi} \quad (\text{I.3})$$

L'option C prend en compte le site entier et implique donc l'utilisation des compteurs et sous-compteurs pour l'évaluation des performances énergétiques de l'entièreté du site. Elle est notamment utilisée lorsque les prévisions d'économies sont importantes en comparaison des facteurs énergétiques non maîtrisés qui influent sur les performances du site, tels que la météo, et l'occupation.

L'option D consiste à étudier un modèle de simulation « calibré » afin qu'il corresponde au mieux aux données mesurées. Elle permet à la fois d'évaluer les économies réalisées sur le site entier à l'instar de l'option C, mais également à un niveau plus fin comme les options A et

B. L'équation I.1 peut être utilisée en utilisant deux versions du modèle : l'un avec les MCE et l'autre sans, les deux étant strictement identiques du point de vue du fonctionnement. I.1 devient alors :

$$\begin{aligned} \text{Economies}_{\text{optionD}} = & \text{Energie de période de référence du modèle calibré sans MCE} \\ & - \text{Energie de période de suivi du modèle calibré avec MCE} \end{aligned} \quad (\text{I.4})$$

Si les données de la période de calibrage sont utilisées à partir de la période de référence, il est important d'ajouter un terme d'erreur sur cette étape de calibrage. I.1 devient alors :

$$\begin{aligned} \text{Economies}_{\text{optionD}} = & \text{Energie de période de référence du modèle calibré sans MCE} \\ & - \text{Energie de période de calibrage (avec MCE)} \\ & \pm \text{Erreur de calibrage dans la lecture du calibrage correspondant} \end{aligned} \quad (\text{I.5})$$

Pour plus de détails sur les différentes options, un tableau récapitulatif se trouve en annexe A. La guideline 14 de l'ASHRAE présente les points techniques d'application de ce protocole.

3.2.3 ASHRAE : Guideline 14-2002

L'ASHRAE³ a pour objectif le développement des sciences du chauffage, de la ventilation, de la climatisation et du refroidissement. Elle permet la création de techniques et de protocoles. Dans le Journal de l'ASHRAE, Rosenberg et Eley ([87], [88]) présentent la directive 14-2002 [9] mise à jour en 2014 [10]. Elle fournit une aide pour mesurer de manière fiable les économies induites par des projets de gestion d'énergie. Elle s'appuie sur le protocole IPMVP et plus particulièrement sur l'option D. La méthode à suivre selon l'ASHRAE est représentée par la figure I.9.

Des détails techniques sont ensuite donnés. Dans le cas d'une simulation calibrée (option D de l'IPMVP), la guideline exige des performances du logiciel de simulation telles que : être capable de modéliser à l'échelle horaire (soit 8 760 heures par an), les effets de masse, l'occupation et les scénarios d'usage, des données météorologiques réelles, etc. Il est également exigé une copie complète des données d'entrée en différenciant celles connues de celles estimées, ces dernières étant affectées d'un degré d'incertitude.

Les données servant au calibrage doivent se baser au moins sur 12 factures d'énergie réparties sur un an (factures mensuelles). Des seuils d'écart sont établis afin de valider le

3. American Society of Heating, Refrigerating and Air-Conditioning Engineers : <https://www.ashrae.org/home>

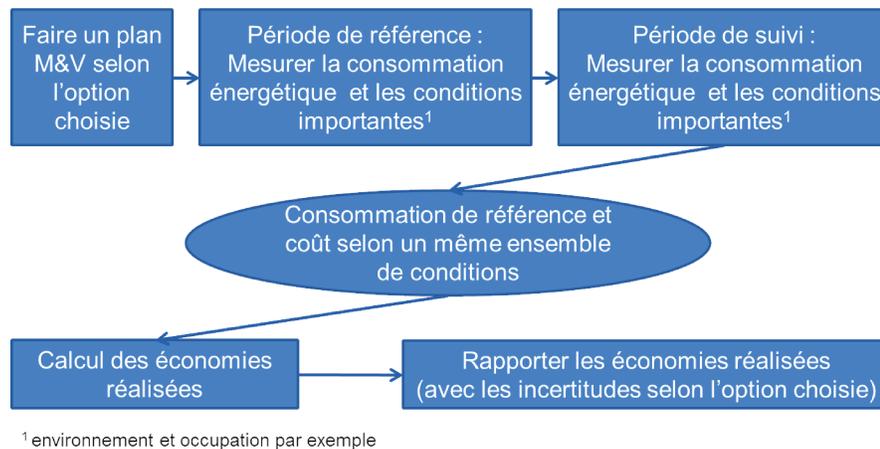


Figure I.9 – Méthode ASHRAE adaptée de ASHRAE [9]

calibrage d'un modèle selon qu'il est effectué sur une base mensuelle ou horaire, en utilisant deux indicateurs d'erreur : la MBE normalisée (??) et le $C_v(\text{RMSE})$ (II.3). Les données ne doivent également pas dépasser des seuils d'incertitude (50% dans le cas d'un rapport annuel, avec un taux de confiance de 68%).

La directive 14 fournit de nombreux détails techniques, notamment sur quelle option choisir selon le cas étudié. Elle permet de mieux se situer et d'appliquer concrètement un plan de M&V.

4 Besoins ressentis et notre proposition pour y répondre △

Dans les parties précédentes, nous avons pu observer que le secteur du bâtiment est un secteur clé si l'on souhaite réduire les consommations énergétiques nationales. Pour cela, le gouvernement a mis en place une réglementation thermique de plus en plus exigeante visant la construction de bâtiments toujours plus performants. De cela, le concept de garantie de performance s'est développé. Nous avons alors exposé sa définition ou plutôt ses définitions. En effet, il existe donc deux garanties de performances : la GPEI, garantie de performances énergétiques intrinsèques, et la GRE, garantie de résultats énergétiques. En France, les CPE, contrats de performance énergétique, ont été mis en place pour répondre à cette thématique. Les CPE se doivent de s'appuyer sur des plans précis pour vérifier que les performances sur site correspondent à celles inscrites dans le contrat. Ici intervient le protocole IPMVP définissant des structures de mesure et de vérification des performances, appelées plans M&V. La directive 14 de l'ASHRAE complète le protocole IPMVP à l'aide de critères techniques.

En parallèle de cela, nous avons observé que les bâtiments neufs, performants, sont souvent équipés de capteurs reliés entre eux grâce à un gestionnaire énergétique permettant à

l'occupant d'optimiser son logement en fonction de son emploi du temps et de ses préférences. Nous avons alors défini le bâtiment intelligent, connecté, tel que nous allons employer le terme dans ce manuscrit : nous nous intéressons ici seulement au fait que les mesures soient disponibles à distance et pas de la gestion du bâtiment par l'occupant.

Les besoins ressentis sont surtout axés autour de la performance énergétique. En effet, les retours d'expériences montrent des écarts importants entre la performance prédite et celle constatée sur site. Il y a donc un besoin d'une méthode permettant d'identifier la cause de ces écarts. De plus, des méthodes existent pour vérifier la qualité du bâti, cependant, elles nécessitent un temps de mesures long, et le déplacement d'une équipe technique sur site accompagné d'un protocole expérimental stricte nécessitant un bâtiment vide, et donc l'absence des occupants.

Il y a donc un besoin d'une méthode permettant d'identifier ces défauts, sans déranger les occupants. C'est ici, que peut servir le bâtiment connecté. Nous disposons grâce à eux d'un grand nombre de données, représentant au minimum la température intérieure et les consommations électriques du bâtiment. Nous avons donc défini une méthodologie, présentée dans le chapitre suivant, permettant d'identifier les défauts (localisation dans le bâtiment et valeur) en utilisant seulement les mesures effectuées sur site et disponibles via une interface externe et le modèle du bâtiment créé lors de l'estimation de ses performances en phase de conception.

Chapitre II

Une nouvelle méthode pour l'identification des défauts

Table des matières

1	Introduction	26
2	L'identification de défauts dans le monde du bâtiment	26
2.1	Etat de l'art sur l'identification de défauts	26
2.1.1	Méthodes d'identification de défauts en cas de présence des occupants	27
2.1.2	Méthodes d'identification de défauts en cas d'absence des occupants	27
2.2	Problèmes liés à ces méthodes	29
3	Principe de la méthode proposée pour l'identification de défauts	30
3.1	Raisonnement pour l'établissement de la méthode	30
3.1.1	Problématique : un écart important entre la phase de conception et celle d'habitation d'un bâtiment	30
3.1.2	Facteurs influençant l'écart entre les deux phases	30
3.1.3	Réduire le nombre de paramètres à étudier	31
3.1.4	Retrouver la cause de l'écart	32
3.2	Description de l'architecture de la méthode	32
3.2.1	Présentation générale	32
3.2.2	Premier bloc : traitement des mesures	32

3.2.3	Second bloc : comparaison des mesures avec les sorties de la simulation . . .	33
4	Choix des techniques pour l'application de la méthodologie.	34
4.1	Identification des périodes d'absence des occupants	34
4.1.1	Etat de l'art de l'identification de la présence des occupants.	34
4.1.2	Choix retenu : analyse des consommations	35
4.2	Prise en compte de l'environnement extérieur et des charges internes.	37
4.2.1	Etat de l'art	37
4.2.2	Choix retenu : création d'un nouveau scénario et d'un nouveau fichier météorologique	39
4.3	Calcul de l'écart entre mesures et simulation : le choix de la RMSE	39
4.4	Identification des facteurs les plus influents : le choix de la méthode de Morris	40
4.4.1	Etat de l'art des différentes analyses de sensibilité	40
4.4.2	Choix pour la réduction du champ d'études : La méthode de Morris	42
4.5	Identification de défauts : le choix de l'algorithme bayésien ABCPMC	45
4.5.1	Etat de l'art des algorithmes bayésiens	45
4.5.2	Choix pour retrouver les valeurs : l'algorithme ABCPMC	52
5	Présentation de l'outil REFATEC (REmote FAult deTEction)	52
5.1	Besoins pour la mise en place de l'outil	52
5.2	Architecture de programmation	53
5.2.1	Traitement des mesures	53
5.2.2	Comparaison entre mesure et simulation	54
5.3	Récapitulatif des développements « outil »	57
6	Conclusion	58
III	Présentation, tests et affinage de la méthodologie par simulation de défauts virtuels	61

Résumé



Nous proposons dans ce chapitre la nouvelle approche développée pour l'identification de défauts. Nous partons du constat établi dans le chapitre précédent que les thématiques du bâtiment connecté et de la garantie de performance peuvent s'allier et aider l'identification de défauts. Nous effectuons donc dans la première partie de ce chapitre un état de l'art pour l'identification de défauts afin de montrer les besoins dans ce domaine. Nous exposons dans la deuxième partie le raisonnement qui nous a permis d'aboutir à notre méthodologie dont nous présentons après l'architecture. Elle se compose de deux blocs principaux : le premier vise à réduire les incertitudes liées à l'occupant et à l'environnement et le second étudie l'écart entre la simulation et la mesure par une analyse de sensibilité couplée à un algorithme bayésien. Dans une dernière partie, nous effectuons un état de l'art des techniques permettant l'exécution de chaque brique de la méthodologie puis nous argumentons notre choix sur une technique spécifique en la décrivant en détails.

1 Introduction △

Nous allons voir dans la première partie de ce chapitre (2) différentes méthodes visant l'identification de défauts dans le bâtiment avec des résultats satisfaisants mais qui nécessitent toutes un travail sur site (tests) avec des protocoles précis demandant ainsi une équipe technique ainsi que l'absence des occupants. Nous sommes partis de ce constat et, en tenant compte du développement du bâtiment connecté que nous avons observé lors du chapitre I, nous allons proposer une nouvelle méthodologie (3) composée de deux blocs principaux : le premier visant à réduire les incertitudes dues à l'occupant et à l'environnement extérieur et le second visant à identifier un défaut. Dans la troisième et dernière partie de ce chapitre, nous effectuerons un état de l'art des différentes techniques à notre disposition pour chaque brique de la méthode. Nous allons alors effectuer un choix et expliciter les techniques choisies pour chaque étape de la méthodologie, menant au schéma final ci-dessous.

2 L'identification de défauts dans le monde du bâtiment △

2.1 Etat de l'art sur l'identification de défauts

Dans le chapitre I, nous avons vu que la problématique de la garantie de performance est au coeur de beaucoup d'études du fait de nombreuses différences observées en termes de confort et de consommation [51]. Nous avons vu que ces différences sont causées par la complexité des systèmes de gestion de l'énergie, par le rejet des occupants de ces systèmes trop contrôlant mais également par une cause amont qui est une mauvaise mise en oeuvre lors de la construction (mauvaises pratiques ou erreur dans les matériaux).

Repérer ces problèmes de mise en oeuvre revient à étudier la garantie de performance énergétique intrinsèque (GPEI) définie au chapitre I.3, qui a fait et fait encore aujourd'hui l'objet de nombreuses études avec notamment le projet PERFORMER [41] et un projet de l'Annex 58 de l'IEA-EBC [85]. Boisson et Bouchié [18] présentent un état de l'art des méthodes pour l'évaluation des performances intrinsèques. Nous en reprenons ici les points principaux. L'étude de la GPEI s'effectue selon deux catégories de méthodes : celles où les occupants sont présents et celles où ils sont absents, procurant donc un bâtiment vide durant une certaine période.

2.1.1 Méthodes d'identification de défauts en cas de présence des occupants

La première catégorie a besoin d'autant de données que possible. En effet, le comportement des occupants étant difficilement prédictible, plus il y a de données, plus les résultats sont représentatifs. La méthode Prism (Fels [46]) présente le premier concept permettant d'étudier la GPEI. Celui-ci consiste basiquement à un bilan énergétique et donc à étudier par exemple la différence de température entre l'intérieur et l'extérieur d'un bâtiment et ainsi d'en déduire le coefficient de perte de chaleur (heat loss coefficient - HLC). Fels insiste toutefois sur le fait qu'il peut être difficile de définir la part du HLC due à l'enveloppe à partir d'informations si globales. Des modèles équivalents de type RC ont ensuite été utilisés afin de définir les caractéristiques thermiques ainsi que celles des systèmes énergétiques d'un bâtiment (Bacher et Madsen [12]). Il est cependant compliqué de relier les paramètres RC aux paramètres physiques du bâtiment tels que la valeur du coefficient de transfert de chaleur (U). D'autres, comme Berger et al. [17], ont pris le parti d'utiliser un modèle de simulation thermique dynamique (STD) et d'y introduire les mesures faites dans le bâtiment et le comportement des occupants, ce dernier étant récupéré grâce à des questionnaires et des interviews.

2.1.2 Méthodes d'identification de défauts en cas d'absence des occupants

La deuxième catégorie vise à minimiser l'impact dû aux occupants, car leur comportement est difficilement identifiable et mesurable. Le test principal existant est le co-heating test (Sonderegger et al. [96], amélioré ensuite par Everett [42]). Bauwens et Roels [13] effectuent un état de l'art sur l'utilisation du co-heating test. Celui-ci consiste, comme pour la première catégorie, à déterminer le HLC d'un bâtiment mais cette fois en traçant la consommation journalière due au chauffage en fonction de la différence entre les températures journalières moyennes intérieure et extérieure. Gorse et al. [49] ont montré que le co-heating test donne une information précise sur le HLC : ils ont effectué deux co-heating tests sur un logement, par deux équipes techniques différentes, mais en suivant la même méthodologie, et ils ont retrouvé la même valeur du HLC à 1% près. Il faut toutefois noter que, bien que ce test soit précis, il présente deux inconvénients majeurs :

- comme il est basé sur la différence de température entre l'environnement intérieur et extérieur, il est plus performant en hiver (car cette différence est plus grande) qu'en été,
- il nécessite des mesures sur une longue durée car il utilise des mesures moyennées sur la journée. Pour avoir un résultat fiable, il faut donc des mesures sur plusieurs jours, voire plusieurs semaines selon les recommandations.

D'autres tests se sont alors développés, toujours dans le cas d'un bâtiment vide, afin de réduire le temps nécessaire pour l'évaluation des performances intrinsèques du bâtiment. Le test QUB (Quick U-Value of Buildings) (Mangematin et al. [65], et Meulemans et al. [68] pour une revue plus complète) et le test ISABELE (In-Situ Assessment of the Building Envelope pErformances) (Boisson et Bougié [18]) permettent de calculer le HLC grâce à des mesures sur quelques jours seulement.

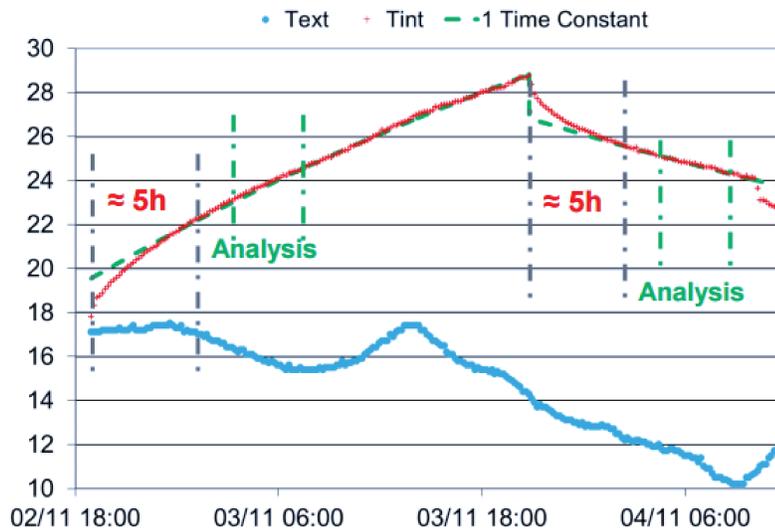


Figure II.1 – Protocole de la méthode QUB - Prandraud [77].

La méthode QUB estime en moins de trois jours la qualité de l'enveloppe (HLC) avec un seuil maximal de $\pm 15\%$. Sur ces trois jours, on chauffe dans un premier temps le bâtiment puis on le refroidit. La figure II.1 schématise la méthode QUB. Pour minimiser les biais dus aux postes de consommation liés à l'usage ou à l'environnement extérieur, ils procèdent comme suit :

- pour les occupants, on travaille sur un bâtiment vide,
- pour la météorologie, on analyse seulement les données nocturnes,
- pour l'électricité spécifique, on mesure tout ce qui ne peut être arrêté,
- et pour la ventilation, on ferme les voies d'aération.

La faisabilité de ce test a été démontrée expérimentalement (Prandraud et al. [79]) puis validée expérimentalement (Prandraud et Fitton [78]) et numériquement (Alzetto et al. [5]).

La méthode ISABELE vise également à estimer la qualité de l'enveloppe (HLC) mais également l'inertie thermique juste après la construction. Contrairement au test QUB, on commence par une période sans chauffage afin d'évaluer la quantité d'énergie stockée par le bâtiment, puis on le chauffe jusqu'à atteindre une consigne de température et enfin, on arrête le chauffage afin d'observer la décroissance de la température intérieure. Un exemple

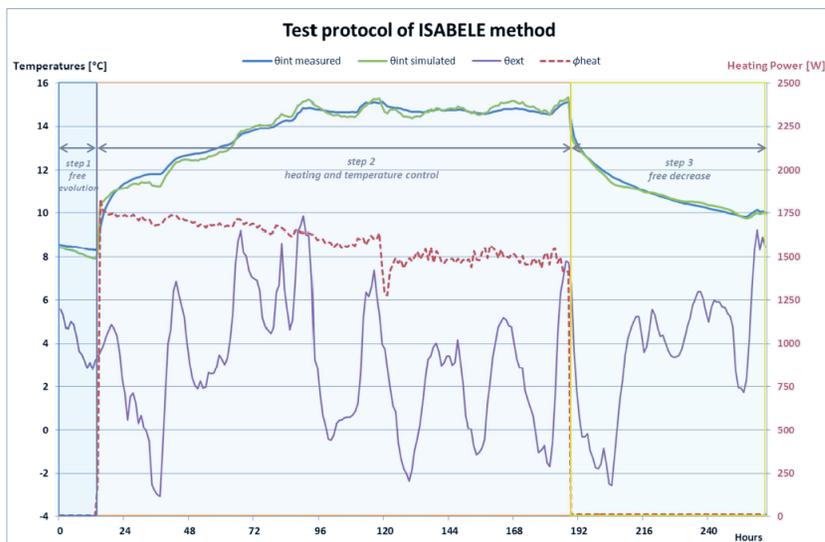


Figure II.2 – Protocole de la méthode ISABELE (Boisson et Bougié [18]).

de protocole est présenté sur la figure II.2. Ces deux méthodes ont été testées dans le cadre du projet PERFORMER [41]. Bouchié et al [21] ont rédigé un rapport où ils testent ces méthodes sur différents cas d'études. Ils en concluent que les deux méthodes sont plutôt efficaces avec une différence inférieure à 10% par rapport au calcul de référence. Ils notent la difficulté de contrôler la quantité de chaleur à apporter pour obtenir une différence de températures suffisante, avec notamment les apports solaires et le stockage de la chaleur dans les parois du bâtiment. Afin de continuer ces recherches, l'AQC [7] (Agence Qualité Construction) présentait en 2016 deux nouveaux projets axés sur la GPEI : le projet MERLIN visant l'élaboration de protocoles de mesures sur site (in-situ) de la performance énergétique intrinsèque (PEI) et le projet EPILOG dont le but est de développer une solution d'évaluation de la PEI de logements.

2.2 Problèmes liés à ces méthodes

Nous pouvons constater que ces méthodes montrent toutes des résultats satisfaisants. Toutes nécessitent pourtant la mise en place d'un protocole bien précis sur site, et donc le déplacement d'une équipe d'experts ainsi que le départ des occupants de leur logement, ou le retardement de leur installation.

A l'époque où les bâtiments deviennent intelligents et connectés (chapitre I), n'y-aurait-il pas un moyen d'utiliser les données issues de ces bâtiments afin d'identifier les défauts intrinsèques au bâtiment pour faciliter à la fois la vie des professionnels du bâtiment et celle des occupants, tout en conservant un temps court de données nécessaires ?

3 Principe de la méthode proposée pour l'identification de défauts

Dans cette partie, nous partons d'abord des thématiques abordées au chapitre I et de la problématique énoncée ci-dessus, puis nous développons le cheminement de notre réflexion en terme d'identification de défauts. Nous expliquons ensuite l'architecture finale de la méthode pour enfin argumenter le choix des techniques que nous employons.

3.1 Raisonnement pour l'établissement de la méthode

3.1.1 Problématique : un écart important entre la phase de conception et celle d'habitation d'un bâtiment

Comme montré au chapitre I, les maisons sont de plus en plus équipées de capteurs et deviennent, par l'intermédiaire de ces derniers, pilotables, ce qui représente un atout non-négligeable pour l'utilisateur du bâtiment. Par exemple, il est alors possible de gérer à distance la température intérieure, l'ouverture ou la fermeture des volets, ou encore l'extinction de l'éclairage. Qui dit capteurs, dit stockage des données. Celles-ci sont situées, dans la plupart des cas, sur des serveurs externes, qui permettent de centraliser les données et de les rendre accessibles aux gestionnaires des données, soit en local soit en ligne. Le fait que ces maisons soient reliées à des serveurs puis aux gestionnaires indique que ces maisons sont connectées. C'est ainsi que sera défini le terme de « maison connectée » pour la suite de ces travaux.

Nous avons également vu que la réglementation thermique exige de plus en plus de performances des nouvelles constructions (1), avec notamment l'horizon 2020, où toute nouvelle maison devra être à énergie positive, tout du moins sur le papier.

C'est là que réside le problème principal : il existe d'importantes différences entre le papier ou autrement dit la phase conception où l'on annonce certaines performances (consommation, production, taux de confort, etc.) et ce qui est réellement ressenti sur place, une fois le bâtiment occupé. De nombreux retours d'expérience montrent que ces nouvelles bâtisses présentent des températures trop élevées ou trop basses, des consommations bien plus grandes que ce qui avaient été prévu, avec des productions plus faibles que celles attendues.

3.1.2 Facteurs influençant l'écart entre les deux phases

Si l'on part du postulat que le modèle est valide, ces différences observées sont alors dues à un ensemble de facteurs très diversifié. On peut séparer ces facteurs en trois catégories :

- les facteurs intrinsèques,
- les facteurs environnementaux,
- les facteurs liés aux occupants.

Les facteurs intrinsèques sont ceux de l'enveloppe du bâtiment, c'est-à-dire tout ce qui concerne les matériaux, les portes, les fenêtres... En d'autres termes, ces facteurs nous indiquent si le bâtiment a réellement été construit selon ce qui a été défini lors de la phase design. Les facteurs environnementaux correspondent aux effets de la météo, de l'environnement extérieur. Ces facteurs sont principalement la température et l'humidité extérieures, le vent, le flux solaire, etc. En effet, lors de l'étude du modèle du bâtiment, nous entrons une météo typique, qui peut ne pas correspondre à celle observée sur site. La dernière catégorie de facteurs représente tous les comportements des usagers qui peuvent influencer les performances du bâtiment. Il s'agit par exemple une augmentation de la consigne de température d'un ou deux degrés, ce qui engendre une consommation excessive d'énergie. Ou plus simplement, d'un usage intensif du four, de la télévision ou de l'éclairage artificiel.

Nous voulons nous intéresser aux facteurs intrinsèques et donc répondre à la question suivante : si l'occupant ressent un inconfort thermique, ou constate une surconsommation, est-ce vraiment dû à sa maison ?

Si inconfort (ou surconsommation) il y a, c'est donc qu'il existe une différence sur la température intérieure (ou la consommation). Il nous est alors paru évident que nous devons étudier cet écart de température (ou consommation).

Cependant, cet écart est influencé par les autres catégories. Une température trop importante peut être due à une météo particulièrement caniculaire cette année-là. Une surconsommation peut être due à des douches très chaudes et/ou très longues. Il apparaît donc compliqué d'étudier seulement une catégorie sans prendre en compte ses deux acolytes : si l'on souhaite étudier la part de cet écart due aux facteurs intrinsèques, nous devons minimiser celle qui incombe aux autres catégories de facteurs, et c'est d'ailleurs ce qui est fait en partie dans le protocole du test QUB (II.2).

3.1.3 Réduire le nombre de paramètres à étudier

Il faut ensuite faire le point sur les différents facteurs intrinsèques. En effet, dans une maison, il existe une multitude de facteurs : les différents matériaux utilisés (et leurs caractéristiques thermiques), les fenêtres (et leurs propriétés), les ponts thermiques, etc. Il est donc nécessaire de trouver une technique qui nous permettra de réduire notre champ d'études aux facteurs qui influencent le plus l'écart observé.

3.1.4 Retrouver la cause de l'écart

Supposons que nous disposons à présent d'un nombre réduit de facteurs potentiellement à la source de l'inconfort (ou de la surconsommation) ressenti par l'occupant, il nous reste alors à identifier notre défaut. Nous nous sommes posés cette question : comment est défini un défaut ? A laquelle nous avons répondu par : il s'agit simplement d'une différence d'un facteur (par exemple l'épaisseur d'isolant). Dans notre cas, nous avons les valeurs théoriques, entrées dans le modèle, et la température et les consommations mesurées au sein du bâtiment qui dépendent des valeurs finales des facteurs. Identifier notre défaut reviendrait alors à identifier ces valeurs finales.

Comme notre étude concerne une maison connectée, nous souhaitons nous restreindre à l'étude des données des capteurs déjà mis en place pour étudier ces facteurs intrinsèques. Cela pose donc notre problématique principale : peut-on détecter des défauts, des malfaçons, seulement en analysant les données de capteurs ? Surtout que celles-ci ne sont pas toujours fiables, car les capteurs ou le système de remontée des données sont parfois sujets à des dysfonctionnements : problèmes d'alimentation, de réseau internet, de positionnement, etc. Nous proposons de répondre à cette problématique dans ces travaux de thèse.

3.2 Description de l'architecture de la méthode

3.2.1 Présentation générale

Nous souhaitons donc développer une méthode permettant de détecter les malfaçons à l'aide de simples capteurs installés dans la maison. Il faut donc trouver des techniques permettant de répondre aux différents points identifiés dans la partie précédente.

L'architecture présentée en figure II.3 a été choisie afin de couvrir les différents points décrits ci-dessus. On peut constater que la méthodologie est divisée en deux blocs principaux : le premier traite les mesures et le second compare les sorties de la simulation (modèle) avec les mesures traitées à l'étape précédente.

3.2.2 Premier bloc : traitement des mesures

Manifestement, la première étape consistera à traiter nos mesures pour réduire l'écart dû aux occupants et à l'environnement.

Pour le premier, nous chercherons à identifier les périodes temporelles où les habitants sont absents. Cela nous permet de limiter au maximum leur impact sur les consommations et sur la température intérieure. S'occuper seulement des périodes d'absence ne suffit cependant

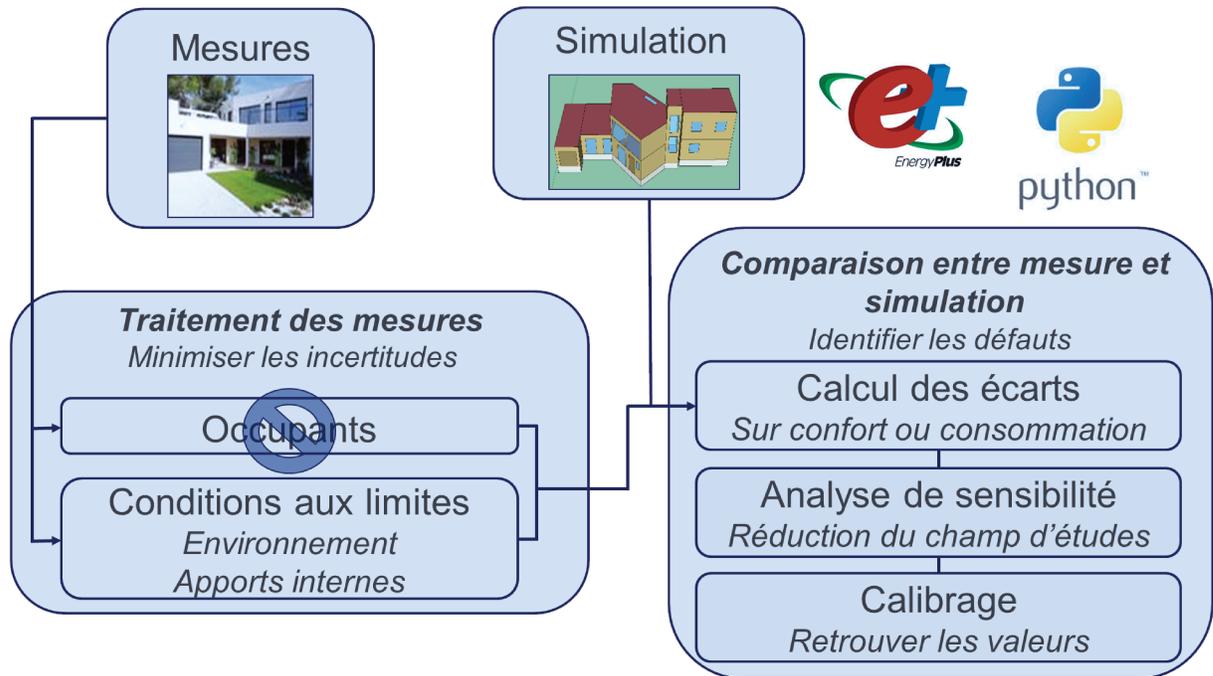


Figure II.3 – Architecture de la méthodologie développée.

pas. En effet, au début de celles-ci, la température et les consommations ont été influencées par le comportement des occupants. Nous devons donc prendre en compte autant que possible leurs usages avant la période d'absence. Pour cela, nous identifierons, parmi nos mesures, les consommations dues à l'électricité spécifique (éclairage, électroménagers, TVs, ordinateurs, etc.). A partir de ces mesures, nous créons un nouveau scénario qui sera implémenté dans notre modèle.

Pour le second, nous créons simplement un nouveau fichier météorologique à partir des mesures dont nous disposons de l'environnement extérieur.

Nous récupérons dans un premier temps les mesures provenant d'un webservice. Grâce à celui-ci, une requête http suffit pour obtenir les mesures qui nous intéressent. Des exemples de requêtes sont disponibles en annexe D. Il est cependant un peu rébarbatif d'entrer manuellement une adresse internet à chaque fois que l'on a besoin de récupérer les mesures. Nous avons donc développé un programme permettant de le faire automatiquement. On s'occupe ensuite des trois catégories de facteurs définies précédemment. En traitant nos mesures, nous identifions les périodes d'absence, les consommations d'électricité spécifique et les données météorologiques.

3.2.3 Second bloc : comparaison des mesures avec les sorties de la simulation

Une fois que notre écart n'est plus lié à l'environnement et aux occupants, il nous faut encore identifier les défauts potentiels. Nous devons tout d'abord faire le tri entre tous les

facteurs intrinsèques du bâtiment (caractéristiques des matériaux et des fenêtres mis en place, infiltrations, ...), afin d'en réduire le nombre et de concentrer notre étude sur ceux ayant le plus de chances d'être la cause de nos défauts. Cela revient à effectuer une analyse de sensibilité, c'est-à-dire que nous allons regarder comment varie notre écart selon différentes valeurs de nos facteurs. Plus la variation est importante, plus les chances que le facteur i cause cet écart sont importantes. L'écart est alors dit sensible à une variation du facteur i .

Lorsque le nombre de facteurs est réduit, nous devons trouver les valeurs qu'ils doivent prendre pour que températures (ou consommations) mesurée et simulée coïncident. En d'autres termes, cela correspond à effectuer un calibrage sur ces facteurs. Le principe général consiste à étudier différentes valeurs de ces facteurs, de regarder pour chaque valeur l'impact sur un certain critère, et d'en conclure de nouvelles valeurs des facteurs étudiés pour que ce critère soit minimisé. Ces valeurs représentent alors ce qui a probablement été mis en oeuvre dans le bâtiment considéré.

4 Choix des techniques pour l'application de la méthodologie

Nous présentons dans cette partie les différents choix scientifiques s'offrant à nous pour répondre aux problématiques exposées ci-dessus. Nous en présentons les avantages et les inconvénients avant de sélectionner celui correspondant le mieux à nos besoins, que nous expliquons finalement.

4.1 Identification des périodes d'absence des occupants

4.1.1 Etat de l'art de l'identification de la présence des occupants

Comme nous avons pu le voir, les occupants sont source de nombreuses incertitudes lors de l'étude thermique du bâtiment. De part leur comportement, ils consomment plus ou moins ce que prévoit le scénario entré dans le modèle du bâtiment. Sidler ([92], [93]) insiste dans plusieurs rapports sur le fait que l'impact des apports internes est de plus en plus important dans les bâtiments nouvelle génération.

Si l'on observe l'impact sur la consommation, Clevenger et Haymaker [26] montrent par modélisation que les gains internes dûs peuvent représenter une surestimation allant jusqu'à 30% sous un climat froid et jusqu'à 20% sous climat chaud. Derouineau et al. [36] mettent en avant les projets européens travaillant sur le bâtiment performants et les résultats obtenus dans chacun des projets. Ils relèvent le rôle récurrent des occupants dans les écarts entre consommations mesurées et estimées.

Pour pallier à cette problématique, nous nous focalisons sur les périodes d'absence des occupants. La présence des occupants a fait et fait toujours l'objet de nombreuses recherches. Page et al. [75] se sont concentrés sur le développement de modèles stochastiques pour la détection de présence des occupants. Ils effectuent en premier lieu un état de l'art des différentes techniques utilisées pour la modélisation de la présence des occupants. Ils énoncent les travaux de Abushakra et al. [1], qui proposent différents profils de gains internes dûs aux occupants par type de bâtiment, ainsi que ceux de Reinhart [82], qui a créé un modèle permettant de gérer l'interaction des occupants avec l'éclairage et les volets. Page et al. appuient les travaux de Wang et al. [103] montrant l'importance des modèles d'éclairage pour la simulation de la présence et l'absence des occupants d'un bureau. Ils introduisent également l'utilisation des chaînes de Markov pour la modélisation de la présence des occupants par les travaux de Yamaguchi et al. [105].

D'autres approches ont été étudiées pour estimer au mieux le comportement des occupants. Jiang et al. [59] utilisent l'apprentissage machine (FS-ELM : feature scaled extreme learning machine) sur le capteur de CO₂ pour estimer l'occupation. Hong et al. [52] présentent les avancées en termes de technologie de capteurs pour observer le comportement des occupants mais également en termes de méthodes analytiques et de modèles. Zhao et al. [106] se concentrent sur l'analyse des données de consommation électrique pour modéliser le comportement des occupants. Vorger explore dans sa thèse [102] une autre approche en développant des scénarios statistiques à implémenter dans le logiciel *Pléiades+Comfie* [57] pour approximer au mieux le comportement des occupants.

Des gestionnaires énergétiques ont été développés afin de mieux contrôler les variables du bâtiment (température, consommation, ventilation) et ils ont prouvé leur efficacité. Ippolito et al. [56] évaluent l'impact de l'automatisation de la gestion énergétique du bâtiment sur une maison individuelle. Ils concluent que ces systèmes de gestion énergétique influent grandement sur la performance énergétique. Pourtant Lowry [64] montre qu'il y a toujours des différences de performance énergétique malgré la mise en place de ces nouveaux systèmes. Il conclut que cela est dû à la non-appropriation de ceux-ci par les occupants.

4.1.2 Choix retenu : analyse des consommations

Notre problématique étant axée sur la détection de défauts et pas sur celle des absences des occupants, et à la vue du temps imparti à ces travaux, nous avons fait le choix de considérer l'identification des absences comme un sujet non prioritaire pour notre problématique qui sera à améliorer à la suite de cette thèse. Nous devons alors trouver une méthode simple, facilement implémentable et donnant des résultats satisfaisants pour le cas d'étude. Il est clair que l'amélioration de cette étape est une perspective importante de ces travaux de

thèse. Notre cas d'études disposant de capteurs, nous avons cherché une méthode, basée sur ces capteurs, simple et rapide permettant d'estimer au mieux si les habitants sont absents.

Nous avons testé deux possibilités : (i) la pose de conditions sur différents capteurs (CO₂, état des lampes et détection de mouvement) et (ii) l'analyse des compteurs des systèmes (chauffage, ventilation et ECS). Nous avons finalement fait le choix de l'analyse des compteurs de consommation des systèmes car les résultats obtenus semblaient plus fiables.

4.1.2. i) Utilisation de capteurs classiques

Dans cette partie, nous étudions deux techniques simples d'identification des absences à partir de mesures menant au choix d'une technique pour la suite. Suite aux résultats, nous choisirons l'une d'elles pour la suite de nos études. Nous utilisons ici les données de la maison POS-A que nous présentons au chapitre III.

Notre première intuition a été d'analyser les capteurs que nous appellerons d'ambiance, soient ceux de CO₂, de détection de présence et ceux de changement d'état des lampes (ON/OFF).

Comme expliqué dans la partie I.4.1, nous voulions un moyen aussi simple qu'efficace pour l'identification des absences. Ainsi, nous avons posé les conditions suivantes présentées dans le tableau II.1. Le tableau II.2 montre les résultats obtenus. Nous observons donc 10 périodes d'hypothétique absence, allant d'une durée de 5h à 3 jours et 11h30.

Capteurs	Condition
Taux de CO ₂	< 500 ppm
Etat des lampes	Changement(1 à 0)
Détection de mouvement	Plateau à 0

Table II.1 – Conditions pour identifier une période d'absence.

N°	Début	Fin	Durée
1	02-01 08 :25	02-01 20 :10	11 :45
2	02-02 07 :50	02-02 21 :20	13 :30
3	02-04 07 :20	02-07 18 :45	3 jours 11 :25
4	02-05 08 :45	02-05 17 :50	09 :05
5	02-06 09 :50	02-06 19 :25	09 :35
6	02-08 07 :25	02-09 18 :35	1 jour 11 :10
7	02-12 09 :30	02-14 20 :45	2 jours 11 :15
8	02-19 11 :00	02-19 19 :40	08 :40
9	02-23 14 :05	02-23 19 :25	05 :05
10	02-29 10 :30	02-29 17 :40	07 :10

Table II.2 – Maison POS-A - Périodes d'absence identifiées grâce à l'analyse des capteurs.

4.1.2. ii) Suivi des consommations électriques

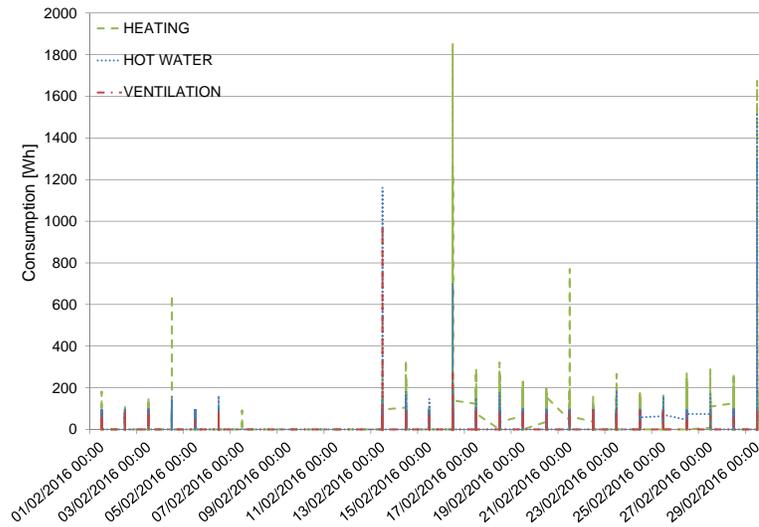


Figure II.4 - Maison POS-A - Consommations des systèmes.

Pour confirmer les résultats issus de l'analyse des capteurs d'ambiance, nous avons analysé les données issues des compteurs électriques des systèmes, dans l'idée que, par exemple, s'il n'y a pas de tirage d'eau chaude, les habitants sont très probablement absents.

La figure II.4 montre les consommations électriques (en Wh) dues à l'ECS, à la ventilation et au chauffage. Nous considérons qu'il y a absence lorsqu'au moins deux de ces consommations sont nulles. On peut aisément remarquer qu'entre le 8 février (matin) et le 13 février (soir), il n'y a aucune consommation liée à ces trois systèmes. On peut donc largement supposer que, sur cette période, les occupants étaient absents.

Le résultat étant plus lisible et marquant dans le cas de l'analyse des consommations, nous utiliserons cette technique par la suite.

4.2 Prise en compte de l'environnement extérieur et des charges internes

4.2.1 Etat de l'art

Cette partie met à niveau notre modèle EnergyPlus en termes de facteurs non liés à la construction même, que nous ne maîtrisons pas, dans le but d'avoir un modèle qui correspond le plus à la situation mesurée.

Dans sa thèse [19], Bontemps présente une méthodologie pour le bon usage des analyses de sensibilité et la fiabilité des outils de simulation. Elle montre que ces derniers sont sujets à de nombreuses incertitudes. Palamo del Barrio et Guyon [76] et De Wit [33], [34] montrent

que les principales viennent de la translation du monde réel au monde informatique, et donc des hypothèses simplificatrices prises en compte. Ensuite, Bontemps [19] présente les différents outils de STD avec leurs avantages et inconvénients, ainsi que les sources d'erreur systématiques que l'on y retrouve. Neymark et al. [73] exposent les différentes erreurs lorsque des différences surviennent entre le bâtiment construit et la phase design.

Bontemps cite notamment Spitz [97] et Goffart [48] qui ont toutes deux travaillé sur les incertitudes au sein d'un bâtiment et notamment sur l'impact des données météorologiques. Spitz [97] montre que selon le fichier météo utilisé des écarts de 30 % pouvaient être constatés sur les besoins de chauffage, les degrés-jours unitaires (DJU) et la moyenne du rayonnement direct. Goffart [48] traite les corrélations que l'on peut avoir entre ces erreurs.

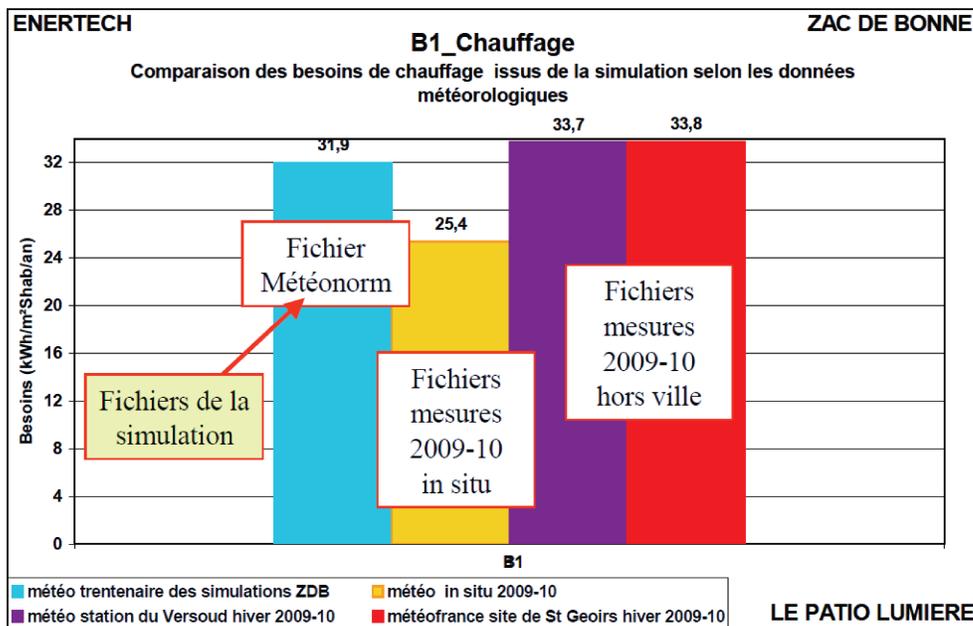


Figure II.5 – Résultats de simulations thermiques dynamiques avec différents fichiers météo - Sidler, [94]

Cela est confirmé par différents retours d'expériences. En effet, dans l'un de ses rapports, Sidler [94] montre que l'utilisation d'un fichier météo d'un autre site (même proche) peut induire une surestimation de la consommation de chauffage de 30%. La figure II.5 présente les résultats de cette étude.

Cette étape correspond donc à la mise à jour du fichier météorologique et des scénarios d'usage de l'électricité (éclairage et usages spécifiques).

4.2.2 Choix retenu : création d'un nouveau scénario et d'un nouveau fichier météorologique

Royapoor et Roskilly [89] présentent une méthode de calibrage d'un modèle *EnergyPlus* grâce aux données météorologiques et aux compteurs d'énergie, ce qui s'approche de notre besoin. En effet, la maison POS-A est équipée d'une station météorologique afin de mesurer au plus près les conditions météorologiques. L'ensemble des capteurs mis en place est présenté au chapitre III, lors de la description du cas d'études.

Nous allons donc créer un nouveau fichier météorologique au format .EPW, compréhensible par *EnergyPlus*, grâce au logiciel *Weather Converter Program*, faisant parti de la suite *EnergyPlus* [39]. Cela nous permet de minimiser l'erreur due à l'écart entre la météo prise en compte par la STD et celle mesurée sur site.

Notre maison mesurant les puissances des appareils électroménagers, des prises et de l'éclairage, nous créons de nouveaux scénarios (Schedule : File dans *EnergyPlus*) basés sur ces mesures.

Ces deux étapes permettent de reproduire le vecteur de charges du modèle de STD, afin de placer au mieux le modèle dans les mêmes conditions que le bâtiment construit.

4.3 Calcul de l'écart entre mesures et simulation : le choix de la RMSE

Les mesures sont faites selon des pas de temps assez proches, environ toutes les 90 secondes. Classiquement, les sorties de simulation se présentent au pas horaire. Afin d'avoir des mesures et les sorties de simulation au même format, nous avons choisi de rester au pas de temps horaire. Celui-ci permet d'avoir une bonne approximation tout en gardant un temps de calculs faible. Avoir des données au pas de temps horaire implique pour chaque variable 8760 valeurs par an et ainsi, 8760 valeurs de notre écart. Il est donc nécessaire de trouver un indicateur fiable permettant d'agréger ces données de sorte à n'avoir qu'une seule valeur par an.

Il existe plusieurs façons de calculer l'erreur (ou différence) entre des résultats de mesures (M_i) et de simulation (S_i). Celle qui semble ressortir est cependant la RMSE (Root Mean Square Error) et la MBE (Mean Bias Error). Diagne et al. [37] les utilisent pour prévoir l'irradiance solaire. Robillart [84] s'en sert comme critère pour la prédiction afin d'améliorer la gestion énergétique des bâtiments performants. L'ASHRAE ([9], [10]) recommande également l'usage de ces deux indicateurs pour le calcul d'écart dans la guideline 14.

$$MBE[\%] = \frac{1}{n} \times \sum_{i=1}^n \frac{S_i - M_i}{M_i} \times 100 \quad (\text{II.1})$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (S - M)_i^2}{n}} \quad (\text{II.2})$$

$$C_v(RMSE)[\%] = \frac{RMSE}{A} \times 100 \quad (\text{II.3})$$

Avec :

- n = Nombre de pas de temps sur la période étudiée ($n = 8760$, lors d'une étude sur un an au pas de temps horaire)
- S = Données simulées
- M = Données mesurées
- $A = \frac{\sum_{i=1}^n M_i}{n}$, la moyenne des mesures

La MBE (II.1) est une mesure adimensionnelle de l'écart entre la variable simulée et mesurée. Elle en permet une vision globale, en calculant la différence moyenne entre la mesure et la simulation. Il faut toutefois noter que cette méthode possède un biais car les écarts positifs sont compensés par ceux négatifs.

La RMSE est donc généralement employée en complément. Celle-ci est une mesure de la variabilité des données. La RMSE est cependant de la même dimension que la variable étudiée. Si l'on étudie la température, la RMSE est alors exprimée de degrés celsius ou kelvin. Le $C_v(RMSE)$ (Coefficient of Variation of the RMSE) représente la RMSE adimensionalisée (II.3). En effet, il est calculé à partir de la RMSE, que l'on divise ensuite par valeur moyenne des mesures sur la période étudiée.

Dans notre cas, nous voulons un indicateur nous permettant une vision globale de notre écart sur toute la période utilisée. Nous calculons, par conséquent, la MBE et le $C_v(RMSE)$.

4.4 Identification des facteurs les plus influents : le choix de la méthode de Morris

4.4.1 Etat de l'art des différentes analyses de sensibilité

Une fois notre écart défini, nous voulons voir quels facteurs l'influencent le plus. Cela va permettre de réduire notre étude d'une centaine de facteurs à une dizaine, ce qui nous permettra à la fois de gagner en précision et de réduire le temps de calcul de la méthode.

Lorsque l'on cherche à identifier les facteurs les plus influents sur un indicateur, il est courant d'utiliser des analyses de sensibilité (AS). Il existe trois catégories principales d'AS : les analyses locales, les analyses globales et les méthodes de criblage. Saltelli et al. [90] procèdent dans leur ouvrage à une description détaillée des différents types d'AS et est d'ailleurs souvent utilisée en tant que référence lorsque les AS sont employées. Plus synthétiquement, un état de l'art a également été effectué par Tian [101].

Saltelli et al. [90] nous informent que les analyses locales consistent à calculer les dérivées partielles de la sortie choisie par rapport aux entrées autour d'un point de référence, permettant ainsi d'identifier l'impact local sur le modèle. Si notre modèle est non-linéaire, les analyses locales montrent toutefois leurs limites. En effet, les indices de sensibilité ne prennent pas la même valeur sur tout l'intervalle d'incertitudes des entrées. Par conséquent, l'information obtenue n'est plus pertinente.

C'est pourquoi, dans le cas d'un modèle non-linéaire, il est recommandé d'utiliser les analyses globales. Celles-ci évaluent l'incertitude globale et la sensibilité de chaque facteur sur la sortie considérée. Elles possèdent deux propriétés principales décrites, encore une fois, dans Saltelli et al. [90] :

- *Inclusion de l'influence de l'échelle et de la forme* : « Les estimations de sensibilité de facteurs individuels prennent en compte l'effet de l'éventail et de la forme de la fonction de densité de probabilité ».
- *Moyenne multidimensionnelle* : « Les estimations de sensibilité de facteurs individuels sont évaluées tout en faisant varier les autres facteurs ».

En ce qui concerne les méthodes de criblage, leur but est d'identifier les facteurs (parmi tous) qui comptent le plus. Elles visent donc à réduire la liste des paramètres importants.

Robillart, dans sa thèse [84], utilise une méthode de criblage afin de réduire son champ d'études pour effectuer un calibrage par la suite. Munaretto [70] combine une méthode de criblage et une méthode d'AS globale afin de valider les modèles numériques. Goffart [48] étudie, grâce à une AS, l'influence des données météorologiques sur les résultats des modèles thermiques. Spitz [97] montre qu'un petit jeu de facteurs prévaut sur tous les autres lors de l'usage d'un modèle de STD.

Iooss [55] effectue une revue des AS et construit un graphique classant les méthodes d'AS selon la complexité du modèle étudié et le nombre d'évaluations de celui-ci. Nous présentons ce graphique en figure II.6. Bien qu'il soit simplificateur, il permet de repérer rapidement les différentes méthodes d'AS selon nos besoins.

Dans notre cas, nous utilisons un modèle complexe (EnergyPlus). Nous nous situons donc dans le cas d'un modèle « non monotone discontinu ». De plus, nous souhaitons pouvoir

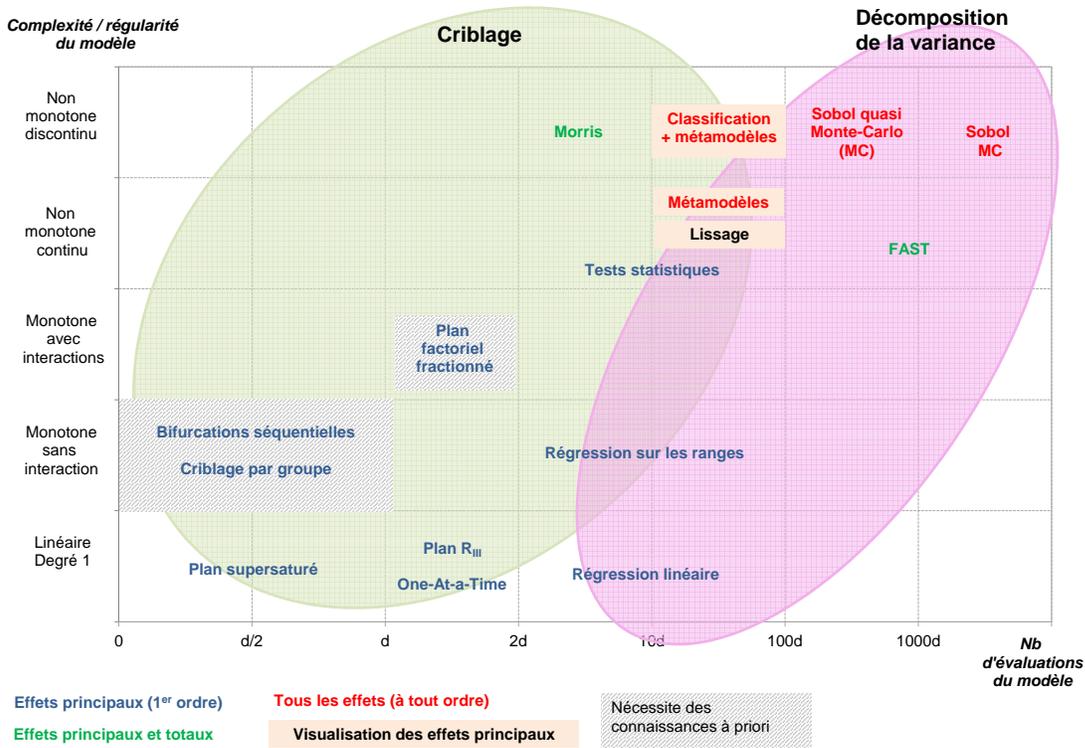


Figure II.6 – Synthèse des méthodes d'AS (coût en nombre d'évaluations du modèle vs. complexité du modèle) - looss [55], adapté.

étudier les différents effets, sans avoir un temps de calcul trop important et donc avec un nombre d'évaluations du modèle faible. Nous avons donc décidé d'utiliser la méthode de Morris que nous présentons ci-après.

4.4.2 Choix pour la réduction du champ d'études : La méthode de Morris

La méthode de criblage selon Morris [69], ou méthode des effets élémentaires, peut se définir comme une « expérience de sensibilité globale » car tout l'espace de variation des facteurs est couvert. Cette méthode est fréquemment utilisée lorsque le modèle considéré coûte cher en termes de temps de calcul ou possède beaucoup de facteurs à étudier. Cette méthode vise à identifier quels facteurs ont :

- des effets négligeables (1)
- des effets linéaires (2)
- des effets non-linéaires ou des effets avec interaction (3).

Le plan d'échantillonnage de la méthode de criblage est décrit de manière assez pédagogique dans la thèse de Maxime Robillart [84]. Nous reprenons donc ici en grande partie son explication. Nous supposons ici N facteurs d'entrée. Nous attribuons à chaque facteur n parmi N une incertitude ($\pm x$), générant ainsi la borne minimale (- x) et la borne maxi-

male (+ x) de son espace de variation. Chaque espace est alors divisé en Q niveaux tels que : $\left\{0; \frac{1}{Q-1}; \frac{2}{Q-1}; \dots; 1\right\}$. Morris répète alors r fois une trajectoire aléatoire passant par $N+1$ noeuds de sorte que chaque facteur ne varie qu'une fois par trajectoire. La figure II.7 illustre trois trajectoires potentielles lorsque N , Q et r sont égaux à 3. Pour maintenir l'uniformité de la distribution des échantillons, Morris définit une longueur de saut Δ telle que $\Delta = \frac{Q}{2 \times (Q-1)}$.

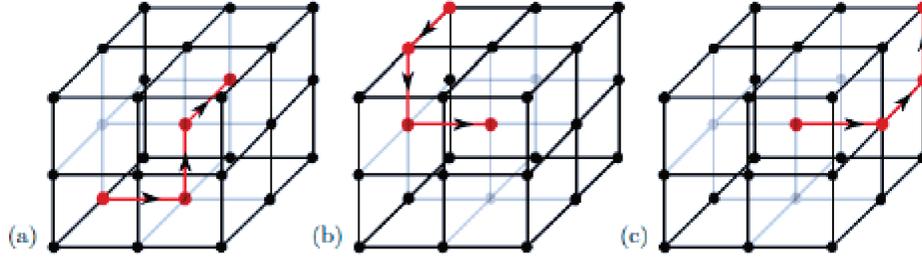


Figure II.7 - Méthode de Morris : exemples de trajectoires, $r = 3$, $N = 3$, $Q = 3$ - Munaretto [70].

La méthode de Morris fonctionne comme suit :

- Pour chaque trajectoire i et pour chaque facteur d'entrée θ_j , on calcule, sur l'espace de variation de θ_j , l'effet élémentaire $E_j^{(i)}$ (II.4) de la sortie M entre deux points consécutifs de la trajectoire.

$$E_j^{(i)} = \pm \frac{M(\theta_1^{(i)}, \dots, \theta_j^{(i)} + \Delta, \dots, \theta_N^{(i)}) - M(\theta_1^{(i)}, \dots, \theta_j^{(i)}, \dots, \theta_N^{(i)})}{\Delta} \quad (\text{II.4})$$

- Deux indices sont alors utilisés : la moyenne μ (II.5) et l'écart-type σ (II.7). Une moyenne élevée signifie que le facteur a une influence importante sur la sortie considérée tandis qu'un écart-type élevé est signe que le facteur a soit des effets en interaction avec d'autres facteurs soit des effets non-linéaires. Afin de contrer les effets des signes opposés, Campolongo et al. [24] définissent l'effet moyen absolu (II.6).

$$\mu_j = \frac{1}{n} \sum_{i=1}^n E_j^{(i)} \quad (\text{II.5})$$

$$\mu_j^* = \frac{1}{n} \sum_{i=1}^n |E_j^{(i)}| \quad (\text{II.6})$$

$$\sigma_j = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (E_j^{(i)} - \mu_j)^2} \quad (\text{II.7})$$

La figure II.8 montre la représentation classique des résultats de la méthode de Morris. Le group (1) englobe tous les facteurs négligeables, le (2) les facteurs avec des effets linéaires et le (3) les facteurs avec des effets non-linéaires ou en interaction avec d'autres facteurs.

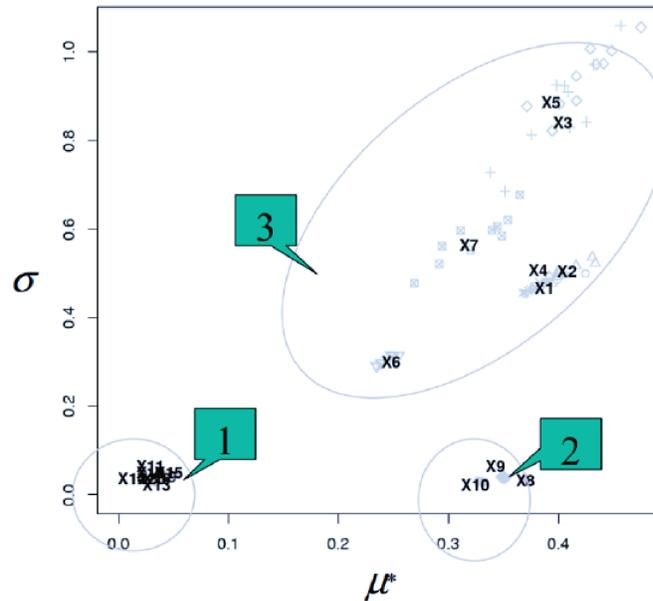


Figure II.8 – Représentation classique des résultats d'une analyse de Morris - looss [54].

Lomas et Eppel [63] sont les premiers à comparer les résultats de différentes AS sur plusieurs logiciels de STD. Plus récemment, Menberg et al. [67] ont comparé les résultats de la méthode de Morris à deux autres AS sur un modèle TRNSYS. Leur conclusion est que les résultats de Morris peuvent être instables soit si un trop grand espace de variation est considéré soit si le paramètre présente un comportement non-linéaire. Ils montrent cependant que 10 trajectoires sont suffisantes pour avoir des résultats fiables, tout en gardant un coût computationnel moindre que les autres AS étudiées. Enfin, ils proposent l'utilisation de la médiane absolue à la place de la moyenne absolue (II.6) afin d'améliorer la robustesse des résultats.

Même si la méthode de Morris présente quelques instabilités, elle correspond à nos besoins. En effet, nous voulions une méthode à faible coût computationnel, permettant d'ordonner les facteurs selon l'importance de leur influence. Nous ferons toutefois particulièrement attention à la robustesse des résultats obtenus, notamment en appliquant l'AS sur au moins 10 trajectoires. Nous pouvons ainsi réduire le champ d'études pour la prochaine étape de la méthode : l'identification des défauts.

4.5 Identification de défauts : le choix de l'algorithme bayésien ABCPMC

Grâce à la méthode de Morris présentée ci-dessus, nous disposons d'un ensemble réduit de facteurs, potentiellement la source de l'inconfort ressenti ou de la surconsommation observée par les occupants du bâtiment. Parmi ces facteurs, il faut encore identifier lesquels ne respectent pas le design initialement prévu. Nous devons donc trouver quelle valeur ces facteurs doivent prendre afin que mesures et simulation concordent au maximum. Nous nous attacherons plus précisément à retrouver leur loi de dispersion, ce qui nous permettra d'avoir un espace de variation pour chaque facteur pour lequel notre critère d'erreur sera respectés.

4.5.1 Etat de l'art des algorithmes bayésiens

Nous utilisons ici un algorithme bayésien. Downey [38] a écrit un livre accessible directement en ligne permettant de comprendre le fonctionnement des statistiques bayésiennes de manière pédagogique. Robert [83] présente un ouvrage plus technique, nécessitant donc une base solide en statistiques, mais complet sur les algorithmes bayésiens.

Ceux-ci s'appuient initialement sur le théorème de Bayes (II.1), dit des probabilités conditionnelles. Comme le décrit Robert [83], ce théorème a été un tournant dans les mathématiques statistiques. Il s'agit en effet de la première inversion des probabilités. Ce théorème cherche donc à exprimer les causes à partir d'observations des effets. L'équation (II.10) exprime que le quotient des probabilités de deux évènements étant donné un effet particulier est égal au quotient des probabilités de cet effet sur les deux évènements. La notation $P(A|B)$ représente une probabilité conditionnelle ou, en d'autres termes, la probabilité d'un évènement A sachant que l'évènement B a eu lieu.

Théorème II.1 (Théorème de Bayes). *Si A et E sont des évènements tels que $P(E) \neq 0$, alors $P(A|E)$ et $P(E|A)$ sont reliés par :*

$$P(A|E) = \frac{P(E|A) \times P(A)}{P(E|A) \times P(A) + P(E|A^c) \times P(A^c)} \quad (\text{II.8})$$

$$= \frac{P(E|A) \times P(A)}{P(E)} \quad (\text{II.9})$$

Et en particulier, si $P(A) = P(B)$:

$$\frac{P(A|E)}{P(B|E)} = \frac{P(E|A)}{P(E|B)} \quad (\text{II.10})$$

Avec :

— A, E et B : des évènements

- A^c : l'évènement complémentaire de A : il se réalise si et seulement si l'évènement A ne se réalise pas.
- $P(A), P(E)$: les probabilités que les évènements A et E se réalisent. C'est donc nos lois a priori
- $P(A|B)$: la probabilité que l'évènement A se réalise sachant que l'évènement B a eu lieu.

En d'autres termes, il permet, pour deux évènements A et E donnés, de déterminer la probabilité de l'évènement A sachant E lorsque les probabilités de A , de E et de E sachant A sont connues. Dans notre cas d'application, il permet par exemple de remonter aux probabilités que nos facteurs prennent telles valeurs à partir de la mesure de la température. Cela revient à la définition suivante :

- A : les valeurs des facteurs intrinsèques de notre modèle, définies par des lois a priori,
- E : la mesure de la température intérieure
- $E|A$: la température intérieure simulée par le modèle avec les valeurs de A ,
- $A|E$: les valeurs des facteurs intrinsèques connaissant la température intérieure mesurée, définies par des lois de dispersion a posteriori.

Pour la suite de l'explication, nous considérerons un jeu de données y et des facteurs θ . Le théorème de Bayes (II.1) permet de trouver la densité a posteriori $\pi(\theta|y)$ si la densité a priori $\pi(\theta)$ et y sont connus. C'est ce qu'on appelle l'inférence bayésienne. Sous forme de formules, on écrit :

$$\pi(\theta) \xrightarrow{y} \pi(\theta|y) \quad (\text{II.11})$$

L'équation (II.11) ci-dessus est équivalente à mettre à jour les incertitudes que l'on peut avoir sur un paramètre suivant des données. $\pi(\theta)$ est ce que « je » crois initialement de θ tandis que $\pi(\theta|y)$ contient ce que « je » pense de θ après avoir pris connaissance de y . De manière schématique, on obtient l'équation (II.12) ci-dessous.

$$\textit{incertitude a priori} \xrightarrow{\text{données}} \textit{incertitude a posteriori} \quad (\text{II.12})$$

Dans la littérature, on emploiera souvent la notation $f(\theta), f(\theta|y)$ lorsque ces lois dépendent d'informations internes aux données. Sinon, on utilise la notation $\pi(\theta), \pi(\theta|y)$.

De nombreuses techniques ont été par la suite développées en s'appuyant sur ce théorème. Wilkinson [104] et Coakley et al [27], en présentent chacun un état de l'art.

Pritchard et al. [81] s'appuient sur les travaux de Tavaré et al [100] pour définir une méthode ABC comme une méthode de rejet permettant de remplacer le calcul de la fonction de probabilité par une simulation de la distribution correspondante.

Le principe des méthodes de rejet repose sur l'établissement d'un critère de rejet, ou seuil d'acceptance. Imaginons que, d'un côté, nous disposons d'un certain nombre de valeurs de nos facteurs θ et, de l'autre, de nos données y alors une méthode de rejet fonctionnera généralement comme suit :

1. on définit un critère d'acceptabilité ϵ ,
2. on évalue le modèle avec les facteurs θ ,
3. on compare la sortie x du modèle, évalué avec θ , avec les données y en calculant une distance d (correspondant souvent à un calcul d'écart, d'erreur)
4. si $d \leq \epsilon$, on accepte la valeur de θ , sinon on la rejette.

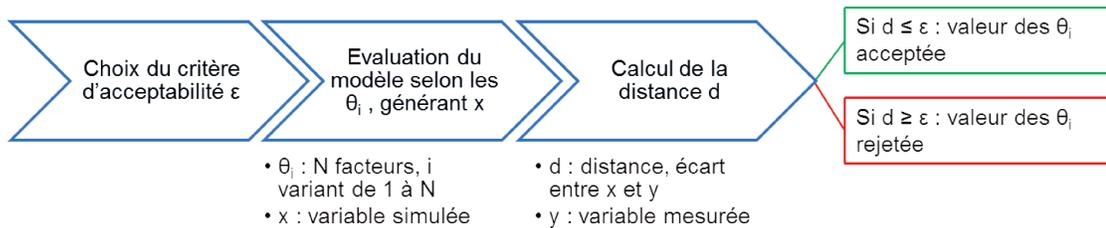


Figure II.9 – Schéma général d'un algorithme de rejet.

Une fois cela établi, différentes améliorations ont été effectuées avec notamment Beaumont et al. [16] qui ont amélioré l'algorithme développé par Pritchard et al [81] en considérant le problème comme un problème d'estimation de la densité conditionnelle, ce qui a permis le développement de techniques autorisant un domaine pour le seuil de rejet plus large, ou encore Marjoram et al. [66] et Borthot et Scisson [20] qui ont augmenté la densité .

Nous nous intéressons particulièrement aux algorithmes ABC-SMC (Sequential Monte-Carlo) introduit par Liu et Chen [62] car ils permettent de tester plusieurs facteurs θ à la fois. Robillart [84] donne dans sa thèse une explication détaillée de ces algorithmes. Nous en reprenons donc ici les points principaux.

L'objectif des algorithmes ABC-SMC est de déterminer N valeurs de facteurs θ satisfaisant le seuil de tolérance critique donné ϵ_c . Pour cela, on part d'un seuil de tolérance maximum ϵ_{max} , puis on effectue une décroissance jusqu'au seuil critique ϵ_c . Pour chaque seuil, une distribution intermédiaire de N_{abc} valeurs de paramètres est déterminée de sorte que le seuil de tolérance soit satisfait. L'espace d'études est alors réduit car l'algorithme se concentre sur l'espace où le facteur θ a le plus de chance de respecter le seuil de tolérance. On se rapproche ainsi de plus en plus de la distribution a posteriori recherchée, grâce à une perturbation puis à un filtrage des facteurs θ .

Sisson et al. [95] montrent cependant que cette approche possède un premier biais. En effet, les valeurs satisfaisant le seuil ϵ_{i+1} n'ont pas été tirées dans la distribution a priori $\pi(\theta)$ mais dans la distribution correspondant au seuil précédent ϵ_i . Sisson et al. décident alors d'associer un poids ω_i à chaque valeur de paramètre. Ce poids ω_i est défini comme la probabilité que la valeur soit tirée de la distribution a priori $\pi(\theta)$ divisée par la probabilité que la valeur soit tirée dans la distribution précédente associée au seuil ϵ_{i-1} .

Malgré la résolution de ce biais, Beaumont et al. [14] montrent qu'il en existe un autre, venant de l'approximation de la distribution a posteriori. Ils ont alors proposé un nouvel algorithme basé sur des populations de Monte-Carlo, nommé ABC-PMC, résolvant ce biais. Nous invitons fortement le lecteur à prendre connaissance de la démonstration faite par Beaumont et al. [14].

Nous présentons dans un premier temps l'algorithme détaillé (1) de cette technique puis un schéma récapitulatif II.10 dans les pages suivantes. Nous rappelons les différentes notations utilisées :

- t : itération variant de 1 à T.
- T : nombre total d'itérations.
- ϵ_t : critère à l'itération t.
- i : index du facteur considéré.
- N : nombre total de facteurs.
- $\theta_i^{(t)}$: facteur i à l'itération t généré à partir de la loi a posteriori trouvée à l'itération t-1. Si t = 1, il correspond à la valeur initiale du facteur.
- x : variable simulée (exemple : température simulée).
- y : variable mesurée (exemple : température mesurée).
- $d(x, y)$: distance, écart entre x et y.
- $\pi(\theta_i^{(t)})$: loi a priori du facteur $\theta_i^{(t)}$.
- $\eta(x|\theta_i^{(t)})$: probabilité d'avoir x avec les valeurs des facteurs $\theta_i^{(t)}$.
- $\omega_i^{(t)}$: poids statistique attribué aux facteurs.
- $\mathcal{N}(\theta_i^*, \sigma_i^2)$: loi normale d'espérance θ_i^* et de variance σ_i^2 .
- \sim : symbole « tel que ».

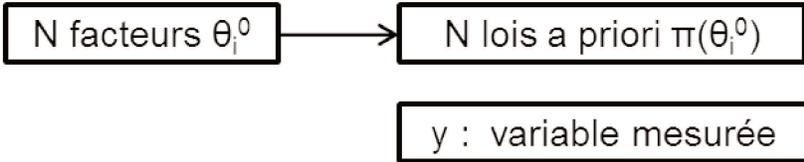
```

for  $t = 1$  do
  for  $i \in [1 : N]$  do
    while  $d(x, y) > \epsilon_t$  do
      | On simule  $\theta_i^{(t)} \sim \pi(\theta_i^{(t)})$  et  $x \sim \eta(x|\theta_i^{(t)})$ ;
    end
    On pose  $\omega_i^{(t)} = \frac{1}{N}$ ;
  end
  On prend  $\sigma_t^2$  comme le double de la variance des  $\theta_i^{(t)}$ 
end
for  $t \in [2 : T]$  do
  for  $i \in [1 : N]$  do
    while  $d(x, y) > \epsilon_t$  do
      | On choisit  $\theta_i^*$  parmi les  $\theta_j^{(t-1)}$  aux probabilités égales à  $\omega_j^{(t-1)}$ ;
      | On génère  $\theta_i^{(t)}|\theta_i^* \sim \mathcal{N}(\theta_i^*, \sigma_t^2)$  et  $x \sim \eta(x|\theta_i^{(t)})$ ;
    end
    On pose  $\omega_i^{(t)} \propto \frac{\pi(\theta_i^{(t)})}{\sum_{j=1}^N \omega_j^{(t-1)} \times \varphi\{\sigma_t^{-1} \times (\theta_i^{(t)} - \theta_i^{(t-1)})\}}$ ;
  end
  On prend  $\sigma_{t+1}^2$  comme le double de la variance des  $\theta_i^{(t)}$ .
end

```

Algorithm 1 : Algorithme ABCPMC.

INITIALISATION :



ALGORITHME :

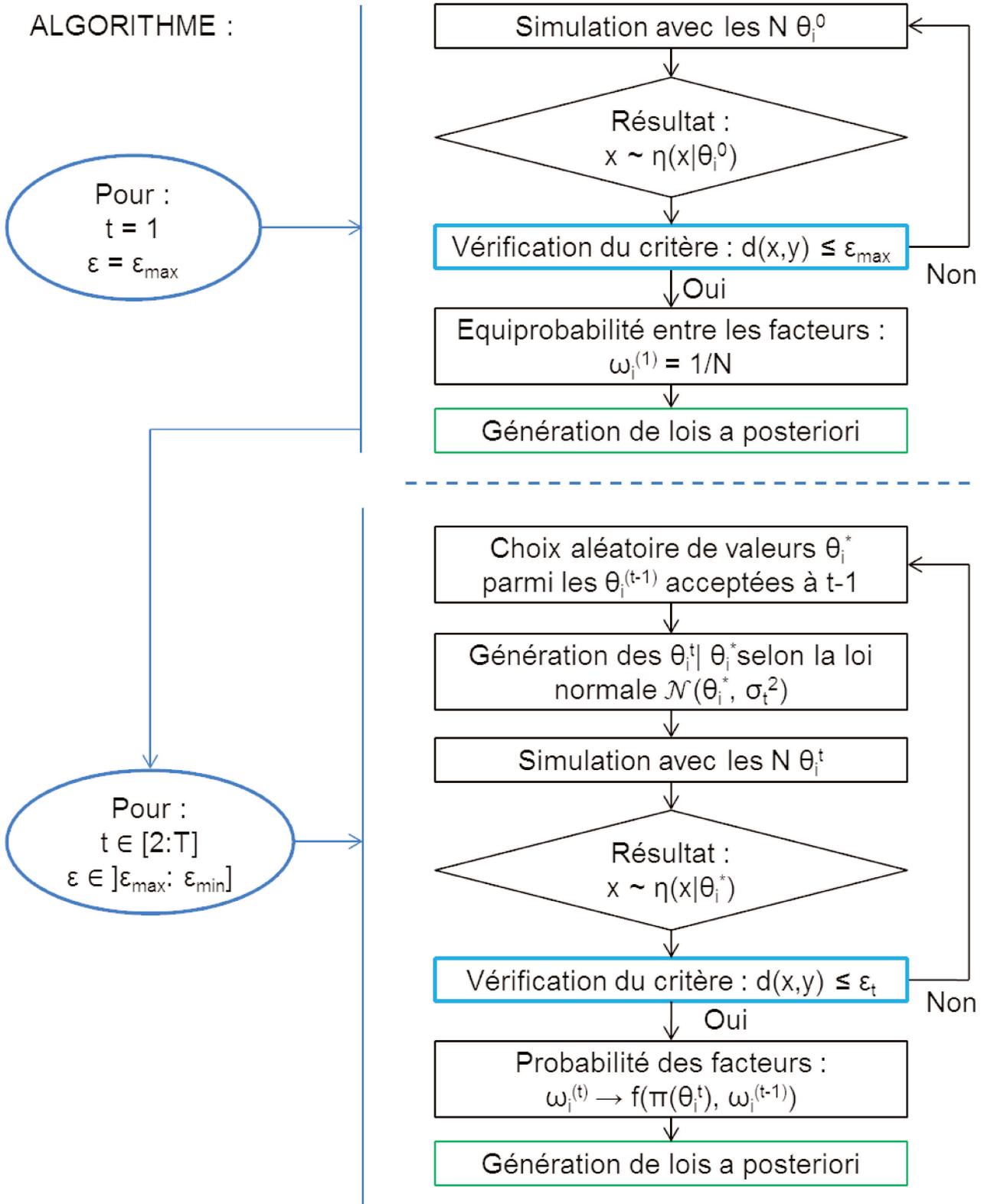


Figure II.10 – Schéma récapitulatif de l'algorithme ABCPMC.

A la première itération sur le seuil ($t = 1$, $\epsilon_{t=1} = \epsilon_{max}$), pour chaque valeur initiale de nos facteurs θ_i , tant que la distance d entre x et y (mesures) est supérieure au seuil maximal ϵ_{max} , on simule notre modèle avec nos lois a priori $\pi(\theta)$ et on considère la sortie x du modèle. A la fin, on attribue le poids ω_i à chaque θ_i .

Pour toutes les autres itérations sur le seuil ($t \in [2 : T]$, $\epsilon_{t=T} = \epsilon_c$), on choisit de nouvelles valeurs des facteurs parmi celles sélectionnées à l'itération précédente, puis tant que la distance d ne respecte pas le seuil ϵ_t , on génère de nouvelles valeurs suivant une loi probabiliste (ici la loi normale \mathcal{N}) et la sortie x du modèle. Une fois le nombre de valeurs acceptées satisfaisant, on attribue un nouveau poids ω_i à chaque θ_i qui dépend de la distribution a priori $\pi(\theta)$ et de la probabilité que la valeur soit tirée dans la distribution associée au seuil précédent ϵ_{i-1} .

Considérons maintenant un exemple comme présenté par la figure II.11. Celle-ci présente différentes lois a posteriori trouvées après une certaine itération et ce afin de visualiser la progression de l'algorithme à différents stades. Pour valider le passage au stade, itération, suivant, l'algorithme doit trouver 1000 valeurs vérifiant le critère associé à l'itération considérée. Dans le cas présenté ici, le critère est la RMSE entre la température intérieure mesurée et celle simulée. Il s'agit donc d'une condition sur l'écart de température. Nous décidons de faire varier ce critère selon une décroissance linéaire de 5 à 0,2°C en 10 itérations.

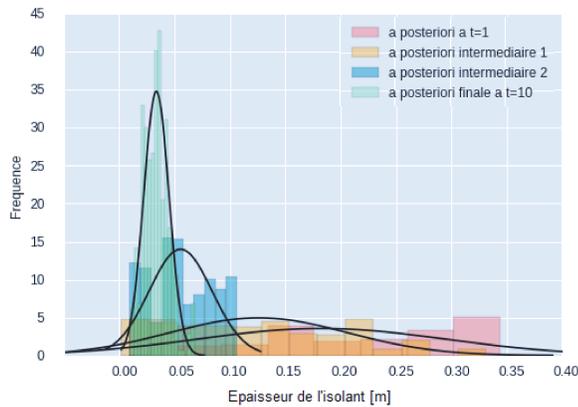


Figure II.11 – Evolution de la distribution a posteriori lors de l'application de l'algorithme ABCPMC

Pour l'initialisation du problème, nous décidons d'étudier l'épaisseur d'isolant d'une maison supposée de 18 cm. Nous représentons celle-ci sous la forme d'une loi normale \mathcal{N} centrée sur la valeur initiale 18 cm (ou plutôt 0,18 m pour pouvoir l'introduire dans *EnergyPlus*) et variant sur un espace défini par l'écart-type correspondant à l'incertitude que l'on souhaite lui affecter. Cette loi est notre loi a priori. Elle est représentée en rouge sur la figure II.11. Pour passer à l'itération suivante, 1000 valeurs doivent vérifier le critère maximal (soit une RMSE inférieure ou égale à 5°C). Suite à cela, une nouvelle loi est générée, appelée loi a pos-

teriori : elle suit le même modèle que la loi a priori, il s'agit donc également d'une loi normale mais elle est cette fois définie par la moyenne des valeurs acceptées pour répondre au critère ϵ et l'écart-type σ de ces valeurs. Cette loi est représentée en orange sur la figure II.11. Une autre itération intermédiaire est représentée en bleu. On observe qu'elles se décalent de plus en plus vers la gauche du graphique qui indique une épaisseur plus faible. Par rapport à la courbe rouge, elles sont centrées sur des épaisseurs inférieures à 0,18 m et ont une incertitude de plus en plus restreinte. Finalement, la dernière loi a posteriori, correspondant au critère de 0,2 °C pour la dixième itération, est représentée en vert sur la figure. On constate qu'elle est centrée sur 0,03 m avec une incertitude réduite. Cela indique que, pour obtenir un écart de 0,2°C entre la température intérieure mesurée et celle simulée, l'épaisseur de l'isolant doit se situer autour de 3 cm.

4.5.2 Choix pour retrouver les valeurs : l'algorithme ABCPMC

Nous avons choisi d'utiliser l'approche présentée par Beaumont et al. [15] : l'algorithme ABC-PMC. En effet, il permet de confronter simulation thermique dynamique et mesures du bâtiment (Coakley et al. [28]) afin d'en évaluer les écarts. Il correspond donc bien à nos besoins, à savoir de trouver les lois de dispersions des facteurs repérés par la méthode de Morris comme influents sur notre écart entre mesures et simulation. De plus, cet algorithme est parallélisable, ce qui permet de réduire considérablement le temps de calcul.

Au niveau de la réalisation, plusieurs choix sont possibles, notamment avec le logiciel *R* (Csillery et al [31], Jabot et al. [58]). Sous *Python*, l'algorithme ABC-PMC est déjà codé et fait l'objet d'un package appelé *abcpmc* développé par Akeret et al. [4].

5 Présentation de l'outil REFATEC (REmote FAult deTECtion)

Nous allons dans cette partie nous intéresser plus grandement à l'outil que nous avons développé pour implémenter la méthodologie proposée précédemment. Nous présentons dans un premier temps les besoins techniques pour sa mise en place puis son architecture de classe et enfin les programmes principaux.

5.1 Besoins pour la mise en place de l'outil

Pour l'exécution de cette méthode, nous ne voulions utiliser que des logiciels libres de droits. C'est pourquoi nous avons décidé d'utiliser EnergyPlus [30] pour la partie simulation thermique dynamique du bâtiment, et *Python2* pour la partie programmation. La méthodologie est composée de deux blocs principaux : la partie sur le traitement des mesures et

celle sur la comparaison entre mesure et simulation. Nous présentons rapidement le premier bloc puis nous nous concentrerons ici sur le deuxième bloc car c'est là que résident les points délicats de programmation.

Celui-ci repose sur l'interopérabilité entre *EnergyPlus* et nos programmes en python. En effet, la méthode de Morris et l'algorithme ABCPMC nécessitent d'évaluer le modèle et donc de lancer *EnergyPlus*. Le besoin primaire a donc été le développement d'un programme permettant de modifier facilement un fichier *.idf* selon les échantillons générés par ces deux méthodes puis d'exécuter *EnergyPlus*.

Pour générer ces échantillons, il faut interagir avec l'utilisateur afin qu'il choisisse les paramètres qu'il souhaite étudier, dans un premier temps pour l'analyse de Morris puis pour l'algorithme ABCPMC. Pour ces deux méthodes, l'échantillonnage est requis sous différentes formes. Il faut donc tenir compte de cela pour la programmation.

Finalement, il nous fallait un code propre à chacune des méthodes utilisées soit un pour la méthode de Morris et un pour l'algorithme ABCPMC.

5.2 Architecture de programmation

5.2.1 Traitement des mesures

Le premier bloc consiste au traitement des données. Nous nous sommes focalisés dans un premier temps sur un pré-traitement en vérifiant qu'elles étaient complètes et cohérentes et dans le cas contraire, nous avons effectué une interpolation linéaire lorsque les données manquantes sont sur une courte durée. Si cette dernière est trop grande (une journée), nous avons dupliqué la journée précédente.

Une fois le pré-traitement effectué, nous avons ré-échantillonné les mesures au format de sortie *EnergyPlus* afin que la comparaison entre mesure et simulation soit plus simple et efficace. Nous avons choisi de reformater les mesures plutôt que la sortie de simulation car cette dernière est calculée à chaque évaluation du modèle et nécessiterait donc un reformatage à chaque fois. Reformater les mesures se fait ainsi une seule fois et permet donc d'économiser du temps de calcul de la méthodologie.

Le programme est nommé « *Resampling.py* » et est disponible en annexe B. La figure II.12 présente les entrées nécessaires ainsi que la sortie du programme avec les principaux packages utilisés.

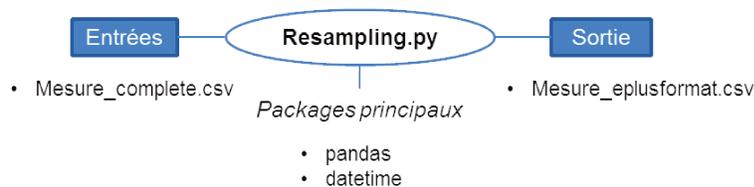


Figure II.12 – Schéma récapitulatif du fichier `Resampling.py`.

Une fois le ré-échantillonnage effectué, nous disposons d'un nouveau fichier csv avec les données au bon format et prêtes à être comparées avec la sortie de simulation.

Nous avons alors simplement tracé les courbes de consommation de nos systèmes afin d'identifier nos périodes d'absence : lorsqu'elles sont à 0, il est fort probable que personne ne soit à l'intérieur d'un bâtiment.

Avant de pouvoir comparer nos mesures avec les sorties de simulation, nous devons encore créer de nouveaux scénarios de consommation due à l'usage afin que notre modèle soit exposé au même niveau de charges internes, avant la période d'absence, que notre bâtiment réel. Nous avons alors créé le code « `ScheduleMesures.py` » (annexe B) permettant de faire cela. La figure II.13 présente son fonctionnement.



Figure II.13 – Schéma récapitulatif du fichier `ScheduleMesures.py`.

Nous avons à présent un fichier *Energyplus* ainsi que des mesures prêts à être comparés. Nous allons maintenant voir comment nous nous y sommes pris en termes de programmation pour effectuer cette partie.

5.2.2 Comparaison entre mesure et simulation

Nous avons exprimé le besoin d'une interopérabilité entre nos méthodes et *EnergyPlus*. La première étape a donc été de créer un programme permettant le lancement d'*EnergyPlus*. Pour cela, nous avons développé un package¹ permettant de facilement lancer une simulation.

1. https://pypi.python.org/pypi/energyplus_wrapper

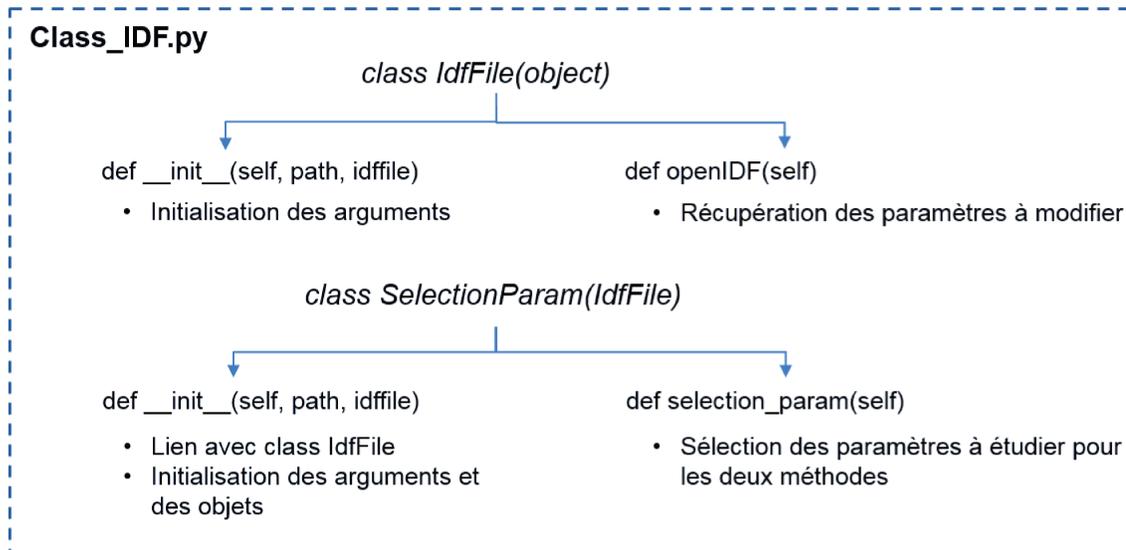


Figure II.14 – Schéma récapitulatif du fichier *Class_IDF.py*.

L'autre point important est la modification des fichiers .idf d'*EnergyPlus*. Comme c'est une fonctionnalité récurrente de la méthodologie, il est nécessaire de travailler par classe afin de pouvoir l'appeler plus facilement. Nous avons créé deux classes regroupées dans le fichier « *Class_IDF.py* » (figure II.14, code en annexe B). La première, « *class_IdfFile* », permet d'initialiser les arguments à utiliser et de récupérer les paramètres qu'il est possible d'étudier. La seconde, « *class_SelectionParam* », hérite de « *class_IdfFile* » afin de récupérer les paramètres puis elle demande à l'utilisateur ceux qu'il souhaite étudier.

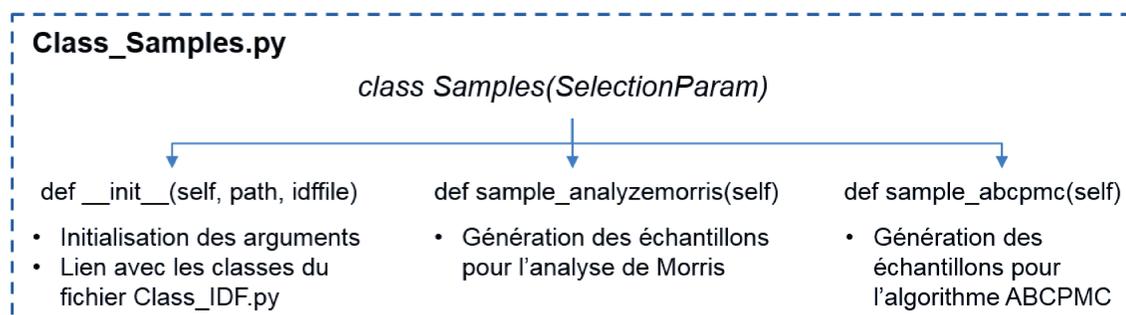


Figure II.15 – Schéma récapitulatif du fichier *Class_Samples.py*.

L'échantillonnage est une autre fonctionnalité commune à la méthode de Morris et à l'algorithme ABCPMC. Nous avons donc créé une nouvelle classe, stockée dans le fichier « *Class_Samples.py* » (annexe B), que nous avons appelée « *class Samples* ». Celle-ci hérite de la classe « *class_SelectionParam* ». Dès qu'un échantillonnage sera demandé, le programme demandera ainsi automatiquement les paramètres sur lesquels il doit être effectué. La classe est composée de trois fonctions : la première initialise le problème en appelant « *class Selec-*

tionParam », la deuxième génère les échantillons au bon format pour l'analyse de Morris et la troisième fait de même mais pour l'algorithme ABCPMC.

Une fois cela effectué, il nous reste à programmer d'une part la méthode de Morris et d'autre part l'algorithme ABCPMC. Il est toutefois recommandé, avant de commencer à programmer n'importe quelle méthode sous Python, de chercher parmi les nombreux packages existants si elles ne sont pas déjà codées. En effet, la communauté des programmeurs Python est de plus en plus grande et regroupe de nombreux domaines d'application. Ainsi, les chances qu'une personne se soit déjà attaquée à des méthodes telles que les analyses de sensibilité ou les algorithmes bayésiens sont très importantes.

En effectuant quelques recherches, nous avons trouvé deux packages : SALib et abcpmc. « SALib »² est une bibliothèque permettant d'effectuer différentes analyses de sensibilité telles que la méthode FAST ou celle de Sobol mais également la méthode de Morris. Nous avons donc utilisé ce package pour notre méthodologie. « abcpmc »³ est comme son nom l'indique dédié à l'algorithme ABCPMC et correspond donc parfaitement à nos besoins.

La figure II.16 présente les principales dépendances du programme effectuant l'analyse de Morris. Elle appelle le fichier « Class_Samples », et demande ainsi à l'utilisateur les paramètres à étudier et génère les échantillons, effectue ensuite l'analyse de Morris grâce au package « SALib » en évaluant notre modèle à l'aide d'«energyplus_wrapper» et traite ensuite les résultats grâce au package « pandas », idéal pour le traitement de données. Le code complet de la méthode est disponible en annexe B.

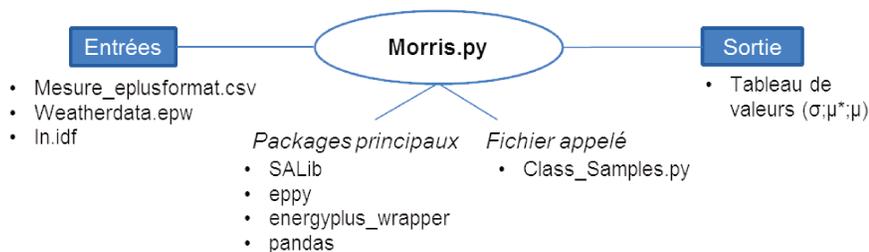


Figure II.16 – Schéma récapitulatif du fichier Morris.py.

La figure II.17 présente à son tour les principales dépendances du programme (annexe B) permettant l'application de l'algorithme ABCPMC. Elle appelle également le fichier « Class_Samples » pour la sélection des paramètres à étudier et l'échantillonnage. Elle fait ensuite appel aux packages « abcpmc » et « energyplus_wrapper » pour le lancement de l'algorithme en lui-même. Le traitement des sorties est effectué grâce à « pandas » et les gra-

2. <http://salib.readthedocs.io/en/latest/>

3. <http://abcpmc.readthedocs.io/en/latest/>

phiques sont générés avec le package « seaborn »⁴, axé sur la création de graphique appliqué aux statistiques.

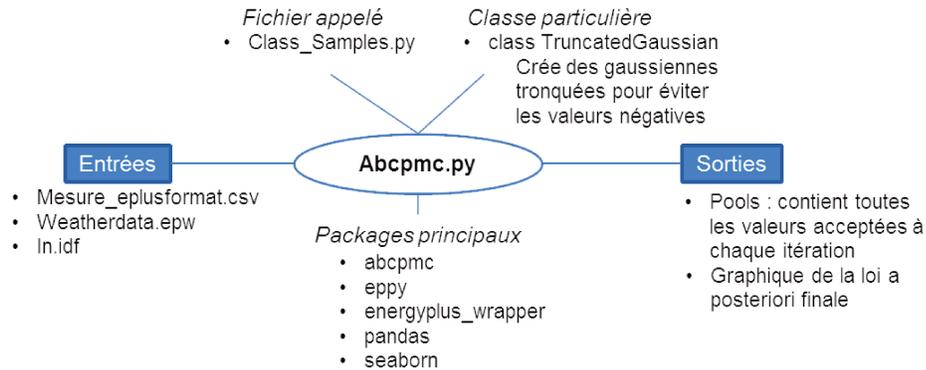


Figure II.17 – Schéma récapitulatif du fichier *Abcpmc.py*.

Ce programme possède quelques subtilités propres à l’application de l’algorithme dans le cas du bâtiment. Nous ne pouvons en effet pas nous permettre d’avoir des valeurs de paramètres négatives (comme une épaisseur d’isolant négative) car cela mène au plantage d’*EnergyPlus*. L’initialisation du problème de l’ABCPMC passe par la définition de lois a priori sous forme de lois normales. Nous avons ainsi créé une classe « class TruncatedGaussian » évitant ces valeurs compromettantes.

L’algorithme ABCPMC nécessite de très nombreuses simulations du fait qu’il produit des résultats stochastiques et donc un temps de calcul long. Nous avons donc parallélisé notre programme afin qu’il puisse lancer plusieurs simulations en même temps sur les différents processus disponibles sur l’ordinateur. Cela a permis de réduire passer le temps de calcul de 8h à 20 minutes pour un faible échantillon. Le package « abcpmc » a été programmé pour facilement implémenter une parallélisation des processus. Pour cela, nous avons utilisé l’objet « Pool » du package « multiprocessing »⁵. Il suffit alors d’indiquer le nombre de processeurs que l’on veut utiliser pour l’algorithme et le package « abcpmc » se charge de répartir les simulations sur ce nombre ci. Il est toutefois à noter qu’il est plus aisé d’utiliser le multiprocessing avec une machine Linux qu’une machine Windows car Linux gère plus facilement et plus rapidement le multiprocessing que Windows.

5.3 Récapitulatif des développements « outil »

Nous nous sommes donc efforcés lors de ces travaux de thèse de proposer un outil, avec un programme simple et efficace, afin de lancer dans un premier temps l’analyse de Morris

4. <https://seaborn.pydata.org/>

5. <https://pypi.python.org/pypi/multiprocessing>

puis l'algorithme ABCPMC dans un second temps. Nous avons créé un package pour faciliter l'exécution d'*EnergyPlus* via *Python*. Le programme pour la méthode de Morris se base sur le package « SALib » et celui de l'algorithme ABCPMC sur le package « abcpmc ». Ce dernier a pu être parallélisé afin de diminuer drastiquement le temps de calcul de la méthodologie.

Le code dans son ensemble se trouve en annexe B et un github est à disposition⁶ afin qu'il puisse être utilisé facilement pour de futurs travaux de recherche.

6 Conclusion

Nous avons vu dans la première partie de ce chapitre (2) différentes méthodes visant l'identification de défauts dans le bâtiment avec des résultats satisfaisants mais qui nécessitent toutes un travail sur site (tests) avec des protocoles précis demandant ainsi une équipe technique ainsi que l'absence des occupants. Nous sommes partis de ce constat et, en tenant compte du développement du bâtiment connecté que nous avons observé lors du chapitre I, nous avons proposé une nouvelle méthodologie (3) composée de deux blocs principaux : le premier visant à réduire les incertitudes dues à l'occupant et à l'environnement extérieur et le second visant à identifier un défaut. Dans la troisième et dernière partie de ce chapitre, nous avons effectué un état de l'art des différentes techniques à notre disposition pour chaque brique de la méthode. Nous avons alors effectué un choix et explicité les techniques choisies pour chaque étape de la méthodologie, menant au schéma final ci-dessous.

6. <https://github.com/rozennjb/REFATEC.git>

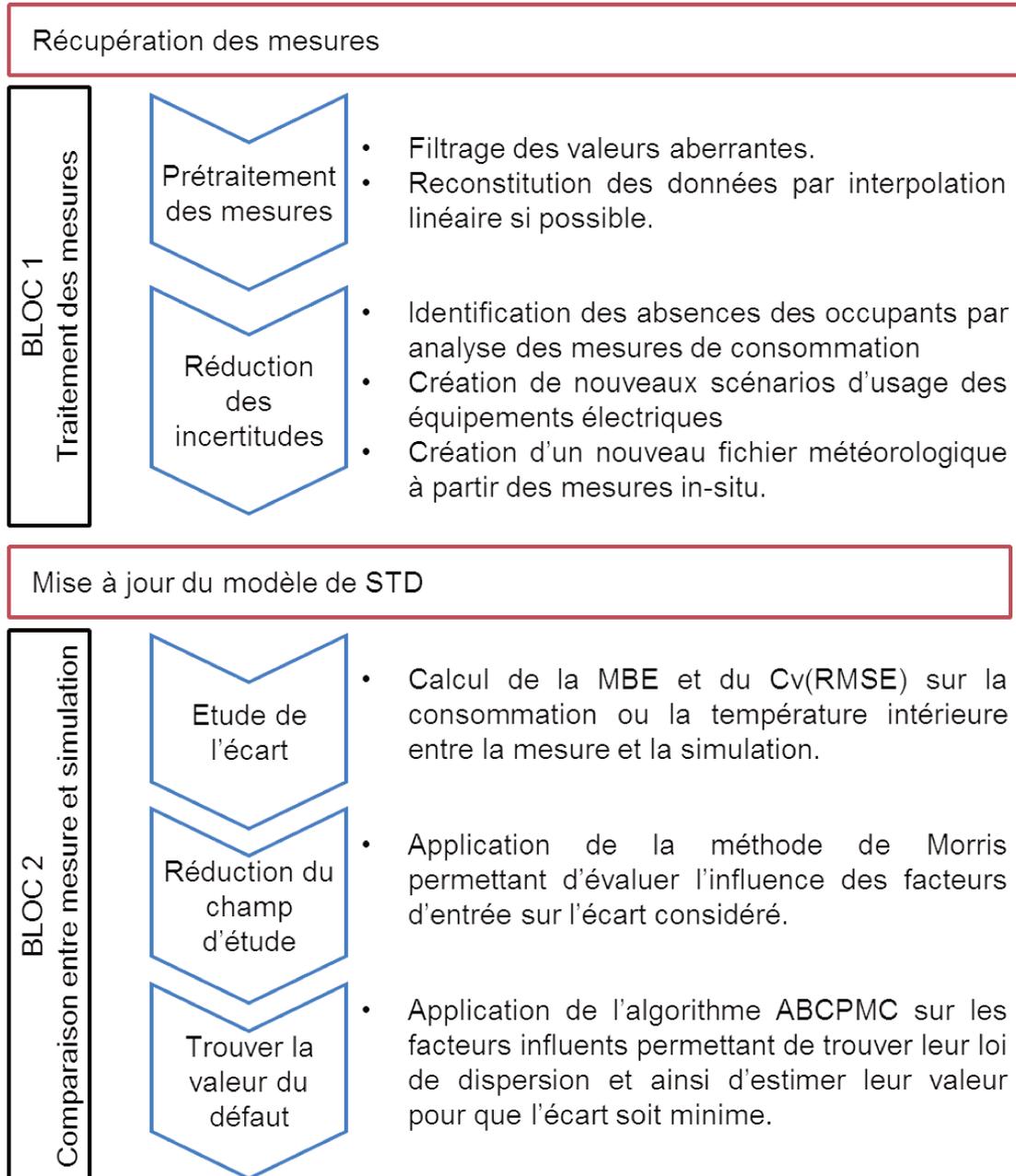


Figure II.18 – Architecture et techniques de la méthodologie développée.

Chapitre III

Présentation, tests et affinage de la méthodologie par simulation de défauts virtuels



Table des matières

1	Introduction	65
2	Présentation du modèle et des hypothèses de simulation prises en compte	66
2.1	Architecture de la maison	66
2.2	Scénarios pris en compte	67
2.3	Défauts à étudier	68
3	Etude de deux indicateurs pour l'identification de défaut	68
3.1	Etude de l'écart de température intérieure entre les deux modèles	69
3.1.1	Fonctionnement des méthodes utilisées appliqué à l'écart de températures	69
3.1.2	Application à la période hivernale	72
3.1.3	Application à la période estivale	74
3.2	Etude de l'écart de consommation électrique due au chauffage entre les deux modèles sur la période hivernale	78
3.2.1	Fonctionnement des méthodes utilisées appliquées à l'écart de consommation	78
3.2.2	Résultat de la méthode de Morris	79
3.2.3	Résultat de l'algorithme ABCPMC	80

3.3	Conclusion sur les indicateurs et l'impact de la saison sur les résultats	81
4	Etude de l'aspect temporel : optimisation du temps de calcul	81
4.1	Durée de simulation optimale.	81
4.2	Durée d'absence optimale	83
4.3	Conclusion sur l'échelle temporelle	85
5	Etude de l'aspect spatial : à quelle échelle arrive-t-on à identifier un défaut?	86
5.1	Défaut minimal détectable par la méthode	86
5.2	Rôle de la durée de l'absence sur la détectabilité du défaut	87
5.2.1	Evolution de l'épaisseur retrouvée selon la durée d'absence et l'épaisseur cible	87
5.2.2	Evolution de l'écart absolu à l'épaisseur cible selon la durée d'absence	89
5.2.3	Conclusion sur le rôle de la durée d'absence sur la détectabilité d'un défaut	91
5.3	Détection d'un défaut global	92
5.3.1	Résultat de l'analyse de Morris dans le cas d'un défaut global	92
5.3.2	Résultat de l'algorithme ABCPMC dans le cas d'un défaut global	93
5.4	Détection d'un défaut local à l'échelle d'une pièce.	93
5.4.1	Mise en place de l'étude	93
5.4.2	Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une pièce.	94
5.4.3	Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une pièce	95
5.5	Détection d'un défaut local à l'échelle d'une paroi	95
5.5.1	Mise en place de l'étude	95
5.5.2	Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une paroi	96
5.5.3	Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une paroi	99
5.6	Conclusion : quelles limites présente notre méthode en terme d'échelle spatiale?	100

6 Typologie de défauts détectables	101
6.1 Défauts introduits	101
6.2 Analyse de Morris pour chaque défaut	102
6.2.1 Défauts repérables	102
6.2.2 Défauts non repérables	104
6.3 Lien entre écart de températures et repérabilité avec l'analyse de Morris.	107
7 Conclusion	109
IV Vers une première mise en situation de l'approche proposée	113

Résumé



Dans ce chapitre nous proposons de tester notre méthodologie dans un cadre théorique maîtrisé. Nous considérons ainsi deux modèles strictement identiques mise à part un défaut inséré dans l'un d'eux. Les variables générées par ce modèle représenteront les données mesurées que l'on aurait dans un cas réel. Le modèle utilisé est présenté dans la première partie de ce chapitre. Dans la deuxième, nous essayons deux indicateurs : l'écart de température intérieure d'une pièce donnée et l'écart de consommation de cette même pièce. Nous appliquons notre méthodologie sur ces deux indicateurs mais également sur deux saisons (hiver et été). Dans la troisième partie de ce chapitre, nous optimisons l'aspect temporel de la méthode en travaillant d'une part sur la durée de simulation et d'autre part sur la durée de la période d'absence. En complément de l'aspect temporel, nous travaillons sur l'aspect spatial dans la quatrième partie de ce chapitre. Nous avons cherché dans un premier temps le défaut minimal que notre méthodologie est capable de détecter. En partant ensuite de ce défaut minimal, nous avons joué sur la durée de la période d'absence pour voir si l'on pouvait encore améliorer la précision de notre approche. Nous avons ensuite créé un défaut plus localisé et avons déterminé jusqu'à quelle échelle la méthodologie développée arrive à le retrouver. Dans la dernière partie de ce chapitre, nous établissons une typologie de défauts en triant parmi ceux qui sont détectés et ceux qui ne le sont pas.

1 Introduction



Dans ce chapitre, nous voulons vérifier si la méthode que nous avons construite fonctionne et si oui, en tester les limites. Pour cela, nous avons décidé de nous concentrer sur un cas particulier, la maison POS-A, située dans le Vaucluse avec donc un climat doux. L'étude se porte sur la comparaison de deux fichiers de simulation : nous comparons donc deux modèles identiques, à l'exception d'un facteur que nous modifierons dans l'un des deux. Ce facteur modifié sera donc le défaut que nous rechercherons. Pour faciliter la compréhension nous nommerons le modèle initial, non modifié, M0 et le modèle modifié M1.

Nous avons choisi la comparaison entre deux modèles comme première approche car seule la simulation permet une évaluation fine de la méthode développée. En effet, nous contrôlons ainsi tous les paramètres ce qui nous offre la possibilité de tout étudier sans perturbation due aux mesures ou à des biais difficilement mesurables ou non identifiés. Cela nous met donc dans des conditions « idéales », non perturbées, de tests que la méthodologie doit passer avant d'être plongée dans des conditions réelles, plus complexes.

L'autre avantage principal est que cela nous donne la possibilité de créer virtuellement n'importe quel défaut permettant ainsi d'étudier la sensibilité de la méthodologie selon différents défauts (important, faible, local ou global), différentes périodes et durées d'absence, etc.

Nous introduisons dans un premier temps un défaut sur l'épaisseur de l'isolant. Au lieu des 18 cm initialement prévus dans M0, nous mettons une épaisseur de 3 cm dans M1. En cherchant ce défaut, nous allons analyser les résultats de l'analyse de Morris qui nous permet de réduire le nombre de facteurs à étudier par la suite en identifiant les facteurs ayant le plus de chance d'être la source de l'écart observé sur notre indicateur. Nous allons étudier l'adéquation de nos indicateurs selon la saison où se situe la période d'absence. En effet, nous nous sommes concentrés sur deux indicateurs majeurs : la température intérieure, liée au confort thermique du ménage, et la consommation énergétique due au chauffage, liée à sa facture énergétique. Ces deux indicateurs sont complémentaires car en été, le chauffage est éteint et nous tenons plus compte de la température intérieure, et en hiver, la température intérieure est souvent pilotée et contrainte par des systèmes de gestion énergétique, surtout lorsque nous considérons des maisons récentes et connectées comme c'est notre cas. Pour ces deux indicateurs et saisons, nous appliquerons ensuite l'algorithme bayésien afin de voir si nous retrouvons le défaut dans tous les cas.

Nous nous posons ensuite différentes questions auxquelles nous essayons de répondre dans les différentes parties de ce chapitre. En premier lieu, nous nous sommes interrogés sur

l'aspect temporel de la méthode. En effet, notre méthodologie repose sur le calcul d'écart sur la période d'absence étudiée. Ceci nous a amené à questionner l'utilité d'une STD sur une année complète. Nous allons donc étudier dans un premier temps la sensibilité du résultat en fonction de la période de simulation. Ensuite, nous avons l'intuition que plus la période d'absence serait longue, plus le résultat serait précis. Nous avons donc décidé d'étudier cela, ce qui nous permettra également de voir la durée limite, la plus courte, nécessaire pour avoir un résultat correct avec notre méthodologie.

Nous nous sommes penchés dans un second temps sur l'aspect spatial de la méthode. En effet, si nous arrivons à détecter un défaut, nous pouvons nous poser la question de la granularité du défaut. En d'autres termes, lorsque nous cherchons un défaut, nous voulons savoir si ce défaut est généralisé au bâtiment ou bien s'il est spécifique à une certaine paroi. Ensuite, il est intéressant et essentiel d'évaluer la performance de la méthode en fonction de l'importance du défaut inséré.

Dans une dernière partie, nous changerons de défaut afin d'évaluer quels types de défauts sont repérables, nous permettant ainsi de créer une typologie de défauts sur laquelle la méthodologie peut être appliquée.

2 Présentation du modèle et des hypothèses de simulation prises en compte

2.1 Architecture de la maison

Le modèle considéré ici (figure III.1) est celui de la maison POS-A, construite au sein du projet COMEPOS¹. Une description du modèle et du bâtiment est disponible en annexe C. Au commencement de ces travaux, la chambre parentale était la pièce comportant le plus de capteurs fonctionnels. L'étude se concentre principalement sur cette pièce dont la composition est décrite par la figure III.2.

Le fait de travailler entre deux fichiers de simulation permet d'imposer les périodes d'absence souhaitées. Ici, deux périodes sont étudiées : une hivernale et une estivale, afin d'évaluer le fonctionnement de la méthodologie sur différentes saisons et sur différents indicateurs. Dans les deux cas, les absences sont créées en associant ces périodes à des périodes de vacances dans le fichier de simulation *EnergyPlus* comme le montre la figure III.3. Les vacances étant considérées comme des jours spécifiques dans le logiciel *EnergyPlus*, définir les absences en

1. <http://www.comepos.fr/>

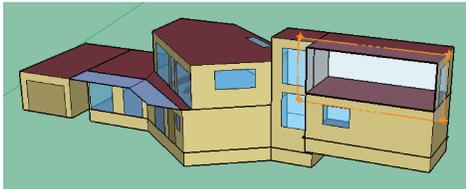


Figure III.1 – Maison POS-A : modèle Sketchup avec coupe sur la chambre parentale

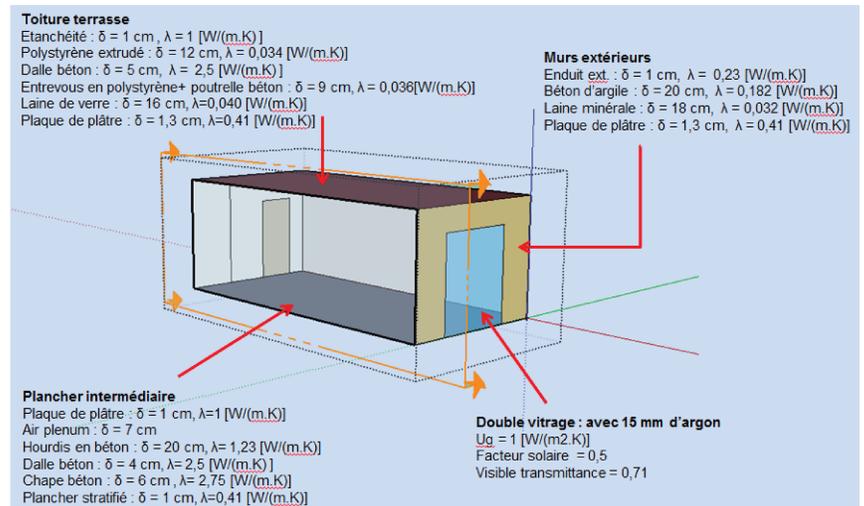


Figure III.2 – Maison POS-A : modèle de la chambre parentale avec composition des parois

tant que vacances permet d'allouer un certain comportement pour chaque scénario pour ces jours précis.

Field	Units	Obj1	Obj2
Name		Vacances_ete	Vacances_hiver
Start Date		10 Aug	08 Feb
Duration	days	5	5
Special Day Type		Holiday	Holiday

Figure III.3 – Simulation des absences dans le logiciel EnergyPlus

En hiver, la période du 8 au 12 février inclus a été sélectionnée, grâce à l'analyse des capteurs (figure II.4), soit une période de 5 jours. Afin d'avoir des résultats soumis aux mêmes conditions, une période de 5 jours a également été choisie pour l'étude de la période estivale : 10 au 14 août inclus.

2.2 Scénarios pris en compte

En hiver, les espaces nuit ont une consigne de température à 18°C avec un réduit à 7°C lors de périodes d'absence prolongée (vacances) et, en été, la température est laissée libre.

En ce qui concerne l'estimation des consommations, elle est basée sur un système idéal piloté par la consigne de température.

2.3 Défauts à étudier

Nous allons nous concentrer dans un premier sur un défaut d'isolation. En effet, c'est le défaut le plus impactant en termes de consommation d'énergie et de température intérieure. Nous avons donc inséré dans notre modèle M1 une isolation avec une épaisseur de 3 cm au lieu des 18 cm présents dans le modèle M0. Les parties 3, 4 et 5 se concentrent sur l'identification de cette différence d'isolant, et donc de retrouver par l'application de la méthodologie une épaisseur d'isolant de 3 cm. Ce défaut est volontairement important afin d'étudier la méthodologie dans un cas extrême. Cela mènera à l'étude de défauts plus faibles.

Dans la partie 6, nous étudions d'autres défauts tels que les performances des vitrages, du béton composant les murs extérieurs et les ponts thermiques.

3 Etude de deux indicateurs pour l'identification de défaut

Comme expliqué précédemment, nous calculons un écart sur deux variables inhérentes à la chambre parentale entre les deux modèles : soit la température intérieure (T_{int}) soit la consommation électrique due au chauffage de la pièce (C_{elec}). Ces écarts sont les indicateurs de notre étude, calculés selon la formule du $C_v(\text{RMSE})$ (II.3).

Variable	Indicateur d'écart pour Morris	Indicateur d'écart (distance) pour ABCPMC	Critères pour ABCPMC
Température T_{int}	$C_v(\text{RMSE}(T_{int}))$	$\text{RMSE}(T_{int})$	$\epsilon \in [5 : 0, 2] \text{ } ^\circ\text{C}$
Consommation C_{elec}	$C_v(\text{RMSE}(C_{elec}))$	$\text{RMSE}(C_{elec})$	$\epsilon \in [250 : 20] \text{ Wh}$

Table III.1 - Variables utilisées et indicateurs d'écart équivalents.

Sur la période d'absence estivale, l'étude sera effectuée seulement sur l'écart de températures car il n'y a pas de consommation due au chauffage sur cette période-ci. Les deux indicateurs seront cependant comparés sur la période hivernale. Il sera alors possible de conclure sur l'indicateur le plus approprié pour l'exécution de la méthode selon la saison étudiée.

3.1 Etude de l'écart de température intérieure entre les deux modèles

3.1.1 Fonctionnement des méthodes utilisées appliqué à l'écart de températures

3.1.1.1 i) Méthode de Morris

La méthode de Morris est basée sur le calcul de l'effet élémentaire d'un facteur sur une certaine variable. Elle requiert un nombre d'évaluation du modèle proportionnel au nombre de facteurs + 1 multiplié par le nombre de répétitions r des trajectoires (III.1). La formule générale (II.4) a été donnée au chapitre 2 lors de l'explication détaillée de la méthode de Morris. L'équation (III.2) ci-dessous représente la formule de l'effet élémentaire appliqué au $C_v(\text{RMSE})$ de la température intérieure T_{int} , soit $C_v(\text{RMSE}(T_{int}))$. Pour faciliter l'écriture et la lecture de l'équation (III.2), la notation $C_{T_{int}}$ est adoptée à la place de $C_v(\text{RMSE}(T_{int}))$.

$$N_S = r \times (n + 1) \quad (\text{III.1})$$

$$E_{j;T_{int}}^{(i)} = \pm \frac{C_{T_{int}}(\theta_1^{(i)}, \dots, \theta_j^{(i)} + \Delta, \dots, \theta_N^{(i)}) - C_{T_{int}}(\theta_1^{(i)}, \dots, \theta_j^{(i)}, \dots, \theta_N^{(i)})}{\Delta} \quad (\text{III.2})$$

Pour chaque facteur d'entrée θ_j , on définit un espace de variation tel que θ_j^i soit compris dans l'espace $[\theta_j - 10\%; \theta_j + 10\%]$. La méthode de Morris calcule ensuite deux indices : la moyenne absolue (II.6)(Campolongo et al. [24]) et l'écart-type (II.7). Les formules (III.3) et (III.4) présentent respectivement la moyenne absolue et l'écart-type adaptés à l'indicateur sur la température $C_{T_{int}}$. Si la moyenne est élevée, le facteur a alors une influence importante, linéaire, sur notre indicateur $C_{T_{int}}$. Si l'écart-type est élevé, le facteur a alors soit des interactions avec d'autres facteurs soit des effets non-linéaires.

$$\mu_{j;T_{int}}^* = \frac{1}{n} \sum_{i=1}^n |E_{j;T_{int}}^{(i)}| \quad (\text{III.3})$$

$$\sigma_{j;T_{int}} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (E_{j;T_{int}}^{(i)} - \mu_{j;T_{int}}^*)^2} \quad (\text{III.4})$$

Pour analyser les résultats de la méthode de Morris, il est donc d'usage de tracer un graphique (σ, μ^*) où chaque facteur est représenté. Il est alors possible, à partir de ce graphique, d'identifier les facteurs ayant le plus d'effets sur notre indicateur $C_{T_{int}}$.

Notre but ici va être de repérer si l'isolant, sur lequel nous avons introduit un défaut, ressort comme facteur très influent. Si oui, il sera alors possible d'appliquer l'algorithme de calibrage, ABCPMC. Si non, il faudra rechercher des explications pour comprendre pourquoi, bien que le défaut inséré soit important, la méthodologie ne le repère pas.

3.1.1. ii) Algorithme ABCPMC

Dans le chapitre 2 (4.5), nous avons détaillé le principe de l'algorithme bayésien ABCPMC, prenant comme base le théorème de Bayes II.1. Nous en rappelons ici les principaux points.

L'ABCPMC consiste à comparer deux vecteurs : la température intérieure simulée (x) et celle mesurée (y), cette dernière correspondant dans ce chapitre à celle générée par le modèle altéré, contenant le défaut de 3 cm sur l'épaisseur d'isolant (2.3). L'algorithme effectue ensuite plusieurs itérations où il va accepter un certain nombre de valeurs d'épaisseurs pour que l'écart de températures satisfasse un critère choisi pour chaque itération.

Considérons maintenant un exemple comme présenté par la figure III.4 . On décide de valider le passage à l'itération suivante lorsque 1000 valeurs pour chaque facteur vérifient le critère suivant une décroissance linéaire de 5 à 0,2°C en 10 itérations (troisième colonne du tableau III.1), le critère initial est donc d'avoir une RMSE inférieure à 5°C et le critère final, une RMSE inférieure à 0,2°C. La première itération teste 1000 valeurs (soit 1000 simulations). Si les 1000 valeurs permettent un écart de température inférieur à 5°C, l'algorithme passe à l'itération suivante, sinon il essaie d'autres valeurs jusqu'à obtenir les 1000 nécessaires. Il reproduit ce schéma jusqu'à la dixième itération où le critère est fixé à 0,2°C.

Pour la première itération, les facteurs étudiés se voient attribuer une loi a priori. Dans notre cas, nous étudions l'épaisseur de l'isolant. La loi a priori de ce facteur est modélisée par une loi normale \mathcal{N} centrée sur la valeur initiale 18 cm (ou plutôt 0,18 mètre pour pouvoir l'introduire dans *EnergyPlus*) et variant sur un espace défini par l'écart-type correspondant à l'incertitude que l'on souhaite lui affecter. La définition de l'écart-type est très importante car, si celui-ci est trop petit, l'algorithme ne pourra pas atteindre le défaut que nous avons inséré, soit une épaisseur de 3 cm au lieu des 18 initiaux, ce qui correspond à un défaut de plus de 80%. A la fin de l'itération, il crée une nouvelle loi appelée loi a posteriori. Elle suit le même modèle que la loi a priori, il s'agit donc également d'une loi normale mais elle est cette fois définie par la moyenne des valeurs acceptées pour répondre au critère et l'écart-type de ces valeurs.

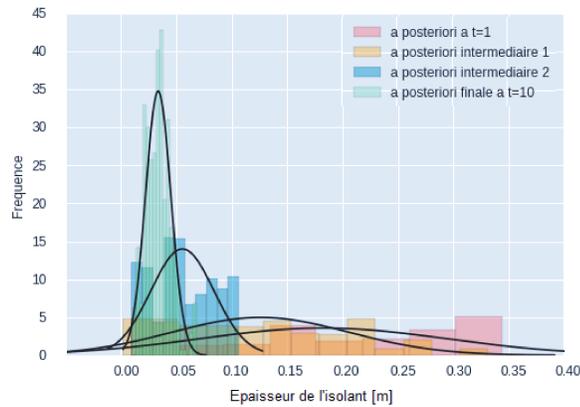


Figure III.4 – Evolution de la distribution à posteriori lors de l'application de l'algorithme ABCPMC.

Pour les itérations suivantes, l'algorithme part de la loi a posteriori trouvée à l'itération précédente et réduit ainsi son espace de variation. C'est ce que montre la figure III.4. La loi a priori est représentée en rouge. Elle est centrée sur 0,18 m avec un espace de variation très large, donc une grande incertitude. Deux lois a posteriori intermédiaires viennent ensuite. La première en orange puis une autre en bleu. On observe qu'elles se décalent de plus en plus vers la gauche du graphique qui indique une épaisseur plus faible. Elles sont centrées sur des épaisseurs inférieures à 0,18 m et ont une incertitude de plus en plus restreinte. La loi a posteriori finale, correspondant au critère de 0,2 °C, figure en vert. On constate qu'elle est centrée sur un résultat proche de celui recherché avec une incertitude faible.

3.1.1. iii) Calcul de l'indicateur d'écart pour l'ABCPMC

Il est à noter que dans le cas de la méthode de Morris, l'analyse des résultats dépend uniquement des indices (μ^* et σ) directement liés à l'indicateur. Afin de pouvoir comparer les résultats selon l'indicateur utilisé (écart de températures ou de consommation), nous avons préféré utiliser le $C_v(RMSE)$ (II.3) qui est adimensionnel plutôt que la $RMSE$ (II.2) dont la dimension correspond à celle de la variable analysée.

Cependant, pour l'algorithme ABCPMC, il est plus parlant de conserver la dimension de cette variable et cela rend la définition des critères plus aisée. De plus, l'analyse des résultats se porte cette fois sur l'épaisseur retrouvée et sur l'incertitude associée. Nous avons ainsi décidé de calculer simplement l'écart avec la $RMSE$ (II.3).

3.1.2 Application à la période hivernale

3.1.2. i) Période d'application

La méthodologie a été appliquée sur la période d'absence hivernale, soit du 8 au 13 février (5 jours), identifiée grâce à l'analyse des consommations des systèmes que nous avons présentée sur la figure II.4 du chapitre II. La simulation est effectuée sur une année complète. L'écart (C_v (RMSE) pour Morris et RMSE pour ABCPMC) est quant à lui calculé seulement sur la période d'absence.

3.1.2. ii) Défaut à identifier

Nous rappelons que nous comparons toujours notre modèle M1, altéré avec une épaisseur d'isolant de 3 cm, et notre modèle initial M0. Dans ces deux modèles, seule l'épaisseur d'isolant a été modifiée.

3.1.2. iii) Résultat de la méthode de Morris

Nous nous attendons ici à ce que notre isolant ressort comme facteur influent, et particulièrement son épaisseur. Il est fort possible qu'en plus de son épaisseur, nous retrouvons également sa conductivité, ces deux facteurs étant liés mais entrés séparément dans *EnergyPlus*.

La figure III.5 présente les résultats suite à l'analyse de Morris. 131 facteurs ont été étudiés (annexe E), les trajectoires ont été répétées quatre fois. 528 simulations *EnergyPlus* ont donc été nécessaires pour l'obtention de ce résultat (équation III.1). Parmi les facteurs étudiés, on retrouve :

- les caractéristiques thermiques de tous les matériaux utilisés dans le modèle (épaisseur, conductivité, densité, chaleur spécifique) ;
- celles des fenêtres (facteur solaire et transmittance) ;
- le taux d'occupation et la puissance des appareils électriques pour vérifier qu'ils n'influencent pas suffisamment l'écart,
- le débit de ventilation,
- les infiltrations.

Chaque cercle représente un facteur. On peut constater qu'une grande majorité se situe en bas à gauche du graphique. Cela signifie que ces facteurs ont un impact négligeable sur l'écart de températures par rapport à ceux situés en haut à droite.

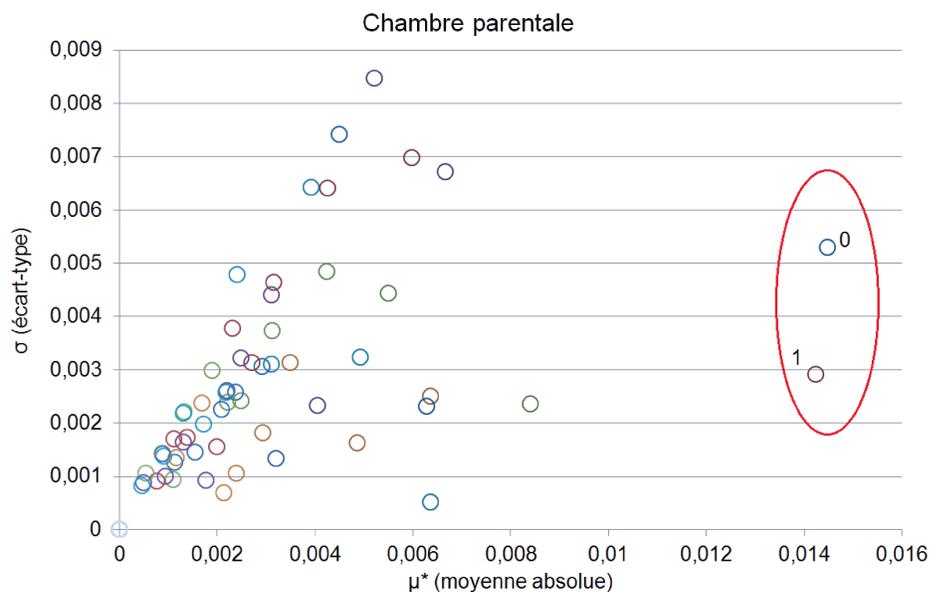


Figure III.5 – Résultat de l'analyse de Morris sur tous les paramètres pour l'échantillon hiver

Les deux cercles à droite, « 0 » et « 1 », représentent la conductivité et l'épaisseur de l'isolant ce qui signifie qu'ils sont, selon l'analyse de Morris, les facteurs ayant l'impact le plus important sur l'écart de température, ce qui est encourageant car l'épaisseur de l'isolant, étant l'élément altéré du deuxième modèle, est le facteur que l'on devait trouver comme important. Ces deux-ci étant les seuls à ressortir de l'analyse de Morris, l'algorithme ABCPMC sera effectué sur ces deux paramètres.

3.1.2. iv) Résultat de l'algorithme ABCPMC

Suite aux résultats de l'analyse de Morris, l'algorithme ABCPMC est appliqué sur l'épaisseur de l'isolant et sur sa conductivité car ces deux facteurs ressortent comme facteur influent. Dans notre modèle M1, seule l'épaisseur de l'isolant a été modifiée et pas sa conductivité ayant une valeur de 0.032 W/(m.K).

Nous avons choisi un critère suivant une décroissance linéaire sur 10 itérations de 5°C à 0,2°C. Le nombre de valeurs à accepter pour chaque itération est de 5000. Il est à noter que l'on retrouve le même résultat si l'on choisit 200 valeurs à accepter, ce qui est très utile si l'on souhaite avoir des résultats rapides sur de nombreuses études. Cependant, nous sommes restés à un nombre de 5000 pour les résultats finaux car ils sont ainsi plus précis du fait que la moyenne et l'incertitude sont directement liés au nombre de valeurs.

La figure III.6 présente le résultat sur l'épaisseur de l'isolant tandis que la figure III.7 montre celui sur la conductivité de l'isolant. Cette dernière représente une gaussienne centrée sur 0,033 W/(m.K), soit une erreur de 3 %. Si l'on regarde maintenant la figure III.6,

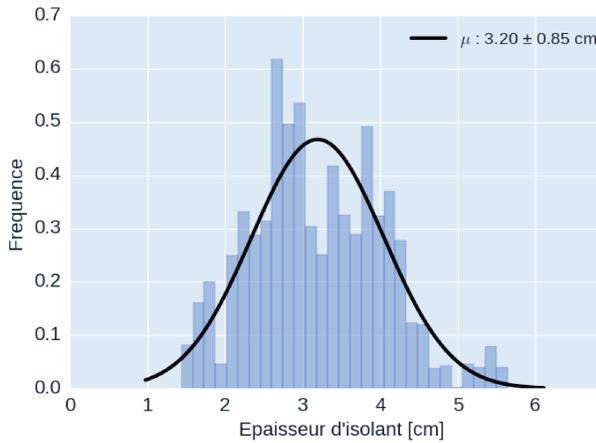


Figure III.6 – Période hivernale - Température - ABCPMC : Résultat de l'algorithme pour l'épaisseur de l'isolant

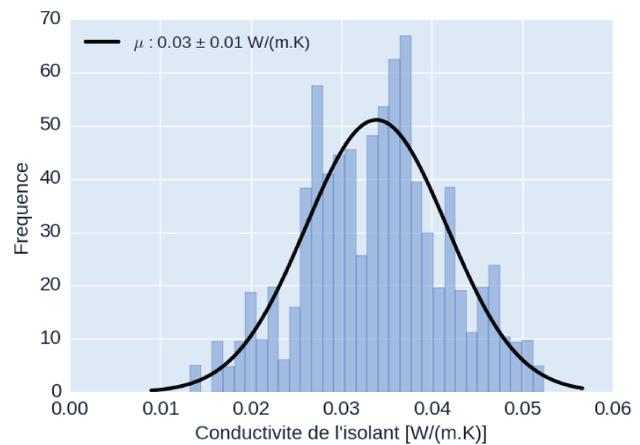


Figure III.7 – Période hivernale - Température - ABCPMC : Résultat de l'algorithme pour la conductivité de l'isolant

représentant le résultat sur l'épaisseur, on observe une gaussienne centrée sur 3,2 cm avec une incertitude de 0,8 cm. Par rapport à l'épaisseur recherchée, 3 cm, on a donc une erreur d'environ 6%.

3.1.2. v) Conclusion sur la période hivernale

Sur cette période, nous avons appliqué la méthodologie développée. Dans un premier temps, l'analyse de Morris a permis de réduire le champs d'études de 131 facteurs initiaux à 2 : la conductivité et l'épaisseur de l'isolant, matériau altéré dont on recherche la nouvelle épaisseur à 3 cm.

Dans un second temps, l'algorithme ABCPMC a permis, pour ces 2 facteurs, de trouver leur valeur la plus probable pour que l'écart entre les températures soit de seulement 0,2°C sur la période d'absence considérée. Pour l'épaisseur, il a retrouvé 3,2 cm (en partant de 18 cm initialement) et pour la conductivité, 0,033 W/(m.K) (au lieu du 0,032 W/(m.K) initial). On en conclut que, pour le cas de la maison POS-A, l'algorithme retrouve le défaut d'isolation dans le cas d'une absence de cinq jours sur une période hivernale.

3.1.3 Application à la période estivale

3.1.3. i) Période d'application

La méthodologie a été appliquée sur la période d'absence estivale, soit du 10 au 15 août (5 jours). La simulation est effectuée sur une année complète. L'écart (C_v (RMSE) pour Morris et RMSE pour ABCPMC) est quant à lui calculé seulement sur la période d'absence.

3.1.3. ii) Défaut à identifier

Nous rappelons que nous comparons toujours notre modèle M1, altéré avec une épaisseur d'isolant de 3 cm, et notre modèle initial M0. Dans ces deux modèles, seule l'épaisseur d'isolant a été modifiée.

3.1.3. iii) Résultat de la méthode de Morris

La figure III.8 présente les résultats suite à l'analyse de Morris sur la période estivale. L'étude s'est portée sur les mêmes facteurs que sur la période hivernale soient 131 facteurs (annexe E) nécessitant 528 simulations *EnergyPlus*.

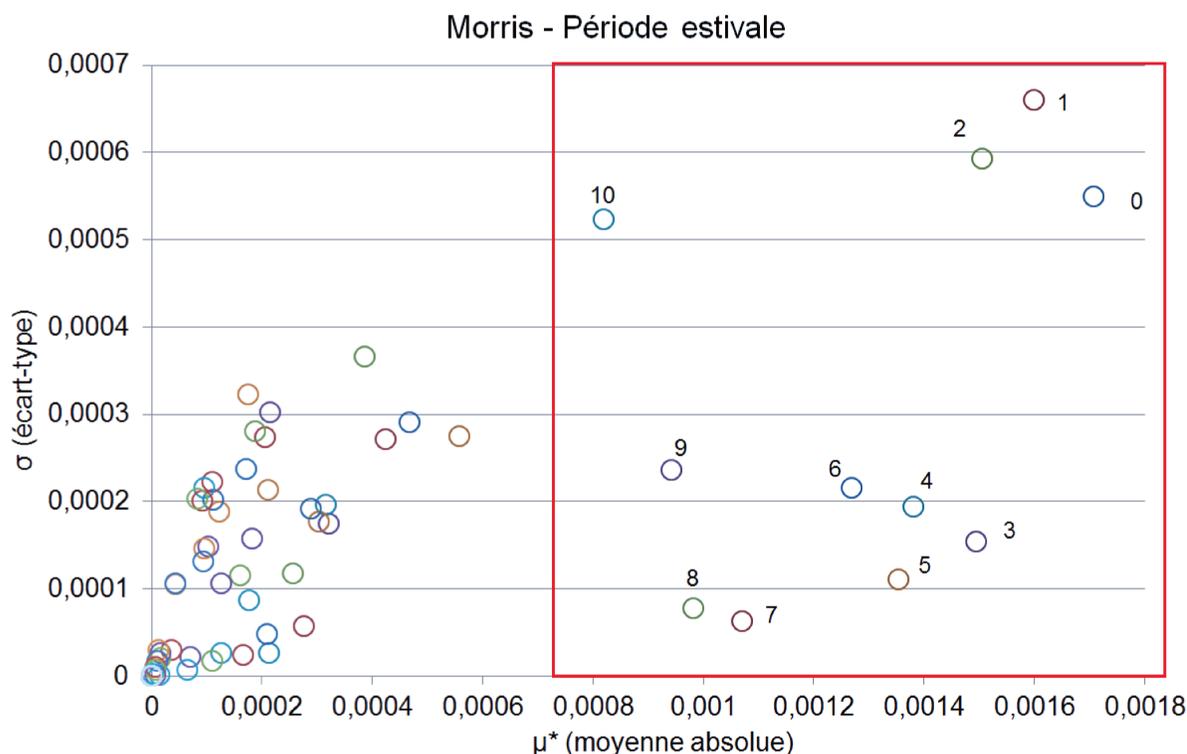


Figure III.8 – Ecart de températures : résultat de l'analyse de Morris sur tous les paramètres pour l'échantillon été

La première chose remarquable est que, comparés à la période hivernale (figure III.5), les cercles, correspondant chacun à un facteur, sont beaucoup plus dispersés, ce qui rend l'application de la méthodologie plus complexe. Ceux importants ont été numérotés et sont répertoriés dans le tableau III.2 ci-dessous, par ordre décroissant de la moyenne absolue μ^* .

On constate que 3 facteurs se démarquent principalement sur la figure III.8 : l'épaisseur de béton d'argile (0) et de nouveau l'épaisseur (1) et la conductivité (2) de notre isolant : la laine minérale.

Index	Facteur
0	Béton d'argile - Epaisseur
1	Laine minérale - Epaisseur
2	Laine minérale - Conductivité
3	Entrevous Polystyrène extrudé - Densité
4	Chape béton - Epaisseur
5	Entrevous Polystyrène extrudé - Epaisseur
6	Chape béton - Densité
7	Dalle béton 4cm - Densité
8	Dalle béton 4cm - Epaisseur
9	Béton d'argile - Conductivité
10	Béton d'argile - Densité

Table III.2 – Légende figure III.8.

Pour les autres facteurs, on peut noter qu'il s'agit principalement de matériaux liés à l'inertie de la maison. En reprenant la figure III.2 présentée au chapitre II, on remarque que : les entrevous sont situés au niveau de la toiture et la dalle et la chape au niveau du plancher intermédiaire.

On peut donc en conclure que, bien que l'isolant soit identifié comme influent, il faut également tenir compte des interactions qu'il peut y avoir, principalement en été, avec la montée en température des parois et donc de l'importance des caractéristiques des matériaux favorisant l'inertie thermique, d'autant plus lors d'un cas à isolation par l'intérieur comme la maison POS-A.

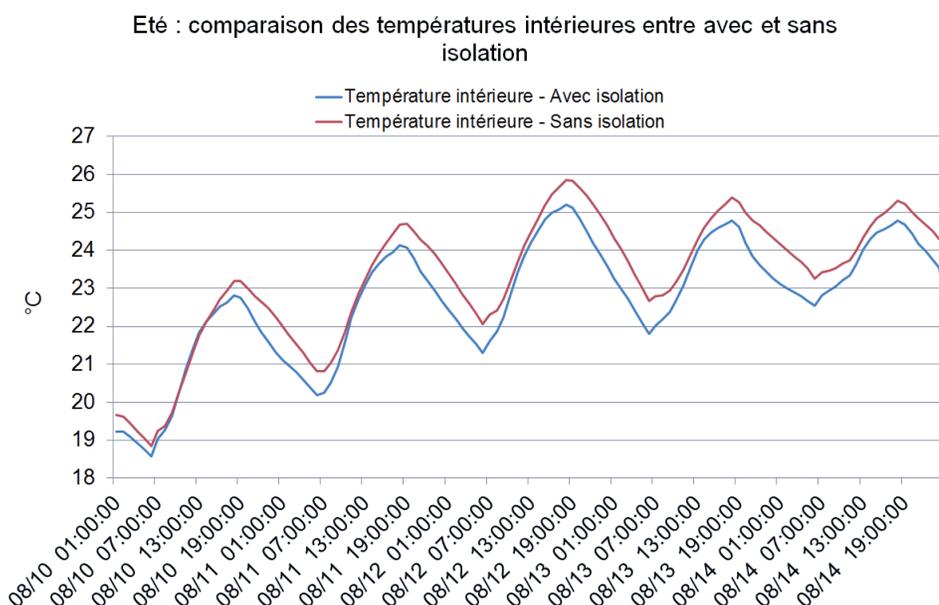


Figure III.9 – Période estivale : comparaison des températures intérieures avec et sans isolation des parois extérieures.

Nous pouvons vérifier cela en regardant les températures intérieures avec ou sans isolant. La figure III.9 montre la différence entre la température avec isolant (bleu) et sans isolant (rouge) sur la période d'absence considérée. On constate que cette différence n'existe quasiment pas : sans isolant, la température est à peine supérieure à celle avec de l'isolant. On voit donc ici que l'isolant n'est pas le seul à influencer la température intérieure mais que l'inertie a un rôle primordial.

3.1.3. iv) Résultat de l'algorithme ABCPMC

Nous allons toutefois appliquer l'algorithme ABCPMC sur l'épaisseur de l'isolant afin de vérifier notre hypothèse qu'en été, l'isolant a moins d'influence, et donc que notre défaut d'isolation devrait être moins identifiable sur cette période d'absence estivale.

Afin d'avoir un aperçu rapide du résultat, 200 valeurs au lieu des 5000 dans le cas précédent, ont été acceptées. Au niveau du seuil, la même définition a été appliquée : une décroissance linéaire de 5 à 0,2°C en 10 itérations. Le résultat est assez éloquent : on trouve une épaisseur de 5,6 cm, avec une incertitude de 2,8 cm, soit 86 % d'erreur par rapport à l'épaisseur recherchée (3 cm).

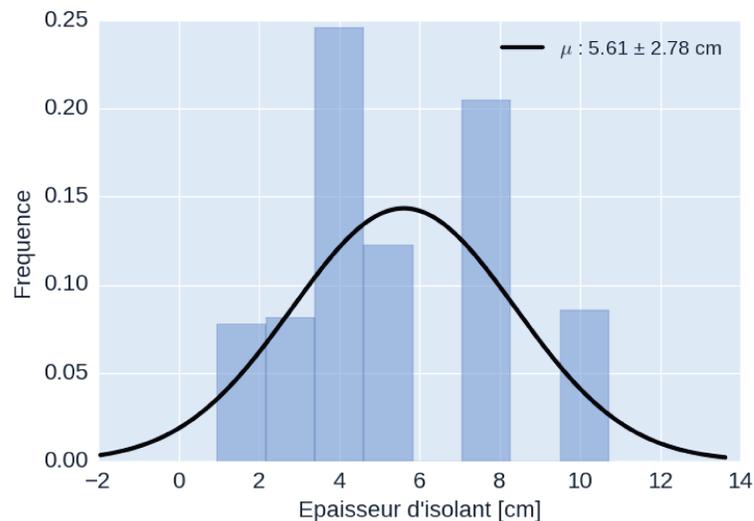


Figure III.10 – Période estivale - Température - ABCPMC : Résultat de l'algorithme pour l'épaisseur de l'isolant *Faire avec 5000 valeurs

3.1.3. v) Conclusion sur la période estivale

Dans cette partie, nous avons appliqué notre méthodologie sur une période estivale. Grâce à l'analyse de Morris, nous avons vu que plusieurs facteurs jouent un rôle important sur l'écart de température : l'isolant a été repéré mais les matériaux à forte inertie également. Le résultat est donc moins frappant que sur la période hivernale, ce qui est problématique pour

l'application de la méthodologie et notamment de l'algorithme ABCPMC, et le résultat de ce dernier a été, comme attendu, moins remarquable que sur la période hivernale : on retrouve une épaisseur de 5,6 cm au lieu des 3 cm recherchés.

Pour ce cas d'études, la période estivale n'est donc pas l'idéal pour l'application de notre méthodologie et il est préférable de l'appliquer sur la période hivernale.

3.2 Etude de l'écart de consommation électrique due au chauffage entre les deux modèles sur la période hivernale

3.2.1 Fonctionnement des méthodes utilisées appliquées à l'écart de consommation

3.2.1. i) Méthode de Morris

Dans cette partie, nous notons C_{chauf} la consommation électrique due au chauffage. Nous calculons toujours le $C_v(RMSE)$ mais cette fois appliqué à C_{chauf} , soit $C_v(RMSE(C_{chauf}))$. Pour simplifier la notation, nous noterons $C_{C_{chauf}}$. La formule générale (II.4) devient alors l'équation (III.5) ci-dessous.

$$E_{j;C_{chauf}}^{(i)} = \pm \frac{C_{C_{chauf}}(\theta_1^{(i)}, \dots, \theta_j^{(i)} + \Delta, \dots, \theta_N^{(i)}) - C_{C_{chauf}}(\theta_1^{(i)}, \dots, \theta_j^{(i)}, \dots, \theta_N^{(i)})}{\Delta} \quad (III.5)$$

Les mêmes indices sont calculés :

$$\mu_{j;C_{chauf}}^* = \frac{1}{n} \sum_{i=1}^n |E_{j;C_{chauf}}^{(i)}| \quad (III.6)$$

$$\sigma_{j;C_{chauf}} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (E_{j;C_{chauf}}^{(i)} - \mu_{j;C_{chauf}}^*)^2} \quad (III.7)$$

Pour analyser les résultats, nous allons toujours tracer le graphique (σ, μ^*) où chaque facteur est représenté, à partir duquel il sera possible d'identifier les facteurs ayant le plus d'effets sur notre indicateur $C_{C_{chauf}}$.

Notre but ici est toujours de repérer si l'isolant, sur lequel un défaut a été introduit, ressort comme facteur très influent.

3.2.1. ii) Algorithme ABCPMC

L'algorithme ABCPMC va fonctionner exactement de la même manière que pour l'écart de températures (3.1.1. ii)) hormis pour la définition des critères. En effet, l'unité traitée est ici des watts et non plus des degrés celsius. Nous utilisons ici toujours une décroissance linéaire sur 10 itérations mais cette fois entre 250 et 25 Wh.

3.2.2 Résultat de la méthode de Morris

La figure III.11 présente les résultats de l'analyse de Morris, où les 131 facteurs ont été traités en 528 simulations *EnergyPlus*. Le facteur (0) ressort principalement : l'épaisseur de la laine minérale, l'isolant que nous avons altéré dans l'un des deux modèles. Les cinq suivants sont présentés dans le tableau III.3 : on retrouve les caractéristiques du béton d'argile (murs extérieurs) ainsi que la dalle de béton (plancher intermédiaire) et l'isolant de toiture, la laine de verre.

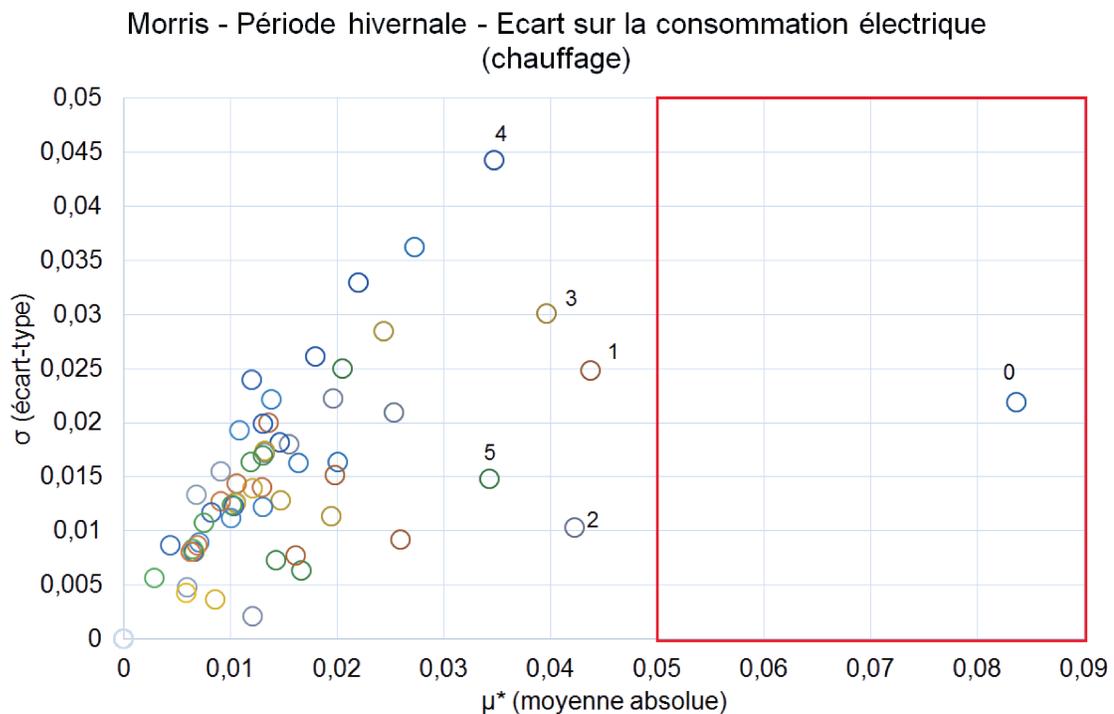


Figure III.11 – Ecart de consommations électriques (chauffage) : résultat de l'analyse de Morris sur tous les paramètres

A la vue de l'important écart qui existe entre l'épaisseur de l'isolant et le reste des autres facteurs, l'algorithme sera appliqué seulement sur celle-ci.

Index	Facteur
0	Laine minérale - Epaisseur
1	Laine minérale - Conductivité
2	Béton d'argile - Epaisseur
3	Laine de verre 16cm - Conductivité
4	Dalle béton 4cm - Epaisseur
5	Béton d'argile - Conductivité

Table III.3 – Légende figure III.11.

3.2.3 Résultat de l'algorithme ABCPMC

Les paramètres pour l'application de cet algorithme sont les mêmes que pour l'étude sur l'écart de températures : 10 itérations sont effectuées en suivant une décroissance linéaire du critère de sélection entre 250 et 25 Wh et nous voulons 5000 valeurs acceptées à chaque itération.

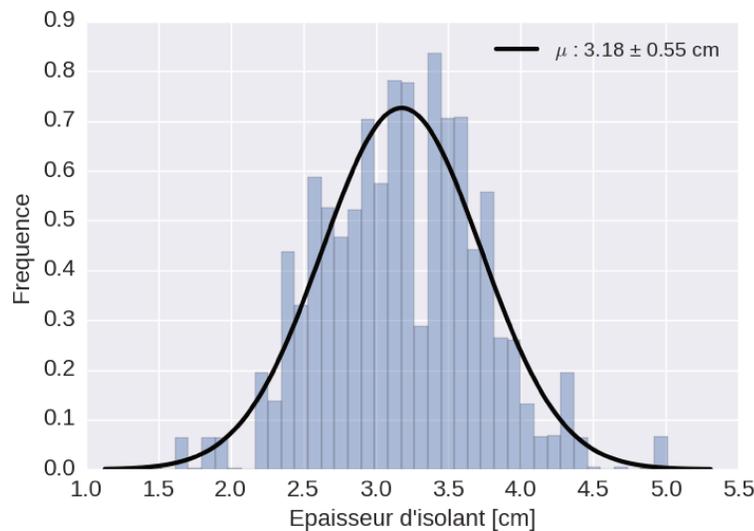


Figure III.12 – Résultat de l'algorithme ABCPMC appliqué à l'écart de la consommation électrique de la chambre parentale

La figure III.12 présente le résultat de l'algorithme obtenu en utilisant 32 process avec un temps de calcul de 2h20. La gaussienne est centrée sur 3,18 cm avec une incertitude de 0,55 cm, soit une erreur de 6%, tout comme avec l'indicateur sur l'écart de températures.

3.3 Conclusion sur les indicateurs et l'impact de la saison sur les résultats

Dans cette partie, nous avons testé deux indicateurs sur deux périodes d'absence différentes, une en été et l'autre en hiver, ce qui permet de voir les limites de la méthodologie en fonction de la saison étudiée.

Lors de l'étude sur la période hivernale, nous avons vu que les deux indicateurs, à la fois l'écart de températures et celui de consommations électriques dues au chauffage, donnent des résultats satisfaisants lors de la recherche du défaut que nous avons engendré : une épaisseur de l'isolant de 3 cm au lieu des 18 initialement prévus. La méthodologie semble donc fonctionner, du moins à cette étape préliminaire, et dans les conditions imposées.

Quant à la période estivale, seul l'écart de températures a été testé et les résultats ont montré que d'autres facteurs influencent la température intérieure, notamment les matériaux à forte inertie. Il n'est donc pas idéal d'appliquer la méthodologie sur une période estivale.

Pour la suite des études, nous conservons la période hivernale pour l'application de la méthodologie. Quant à l'indicateur, les deux semblent donner des résultats corrects. Nous choisissons donc arbitrairement de nous focaliser sur l'écart de températures.

4 Etude de l'aspect temporel : optimisation du temps de calcul



4.1 Durée de simulation optimale

Le choix de la période de simulation est critique pour notre méthodologie. En effet, celui-ci est déterminant pour le temps de simulation. Chacune des deux méthodes utilisées (la méthode de Morris et l'ABCPMC) se basent sur un nombre d'échantillons important, et sur une évaluation du modèle pour chaque échantillon. Ainsi le temps d'évaluation, et donc de simulation, devient crucial si l'on souhaite optimiser le temps nécessaire à l'application de la méthode.

Trois périodes ont été testées :

- Cas 1 : une année entière de simulation, qui est la durée classique d'une simulation thermique dynamique,
- Cas 2 : une période commençant 30 jours avant la période d'absence et finissant à la fin de celle-ci,
- Cas 3 : la période d'absence repérée.

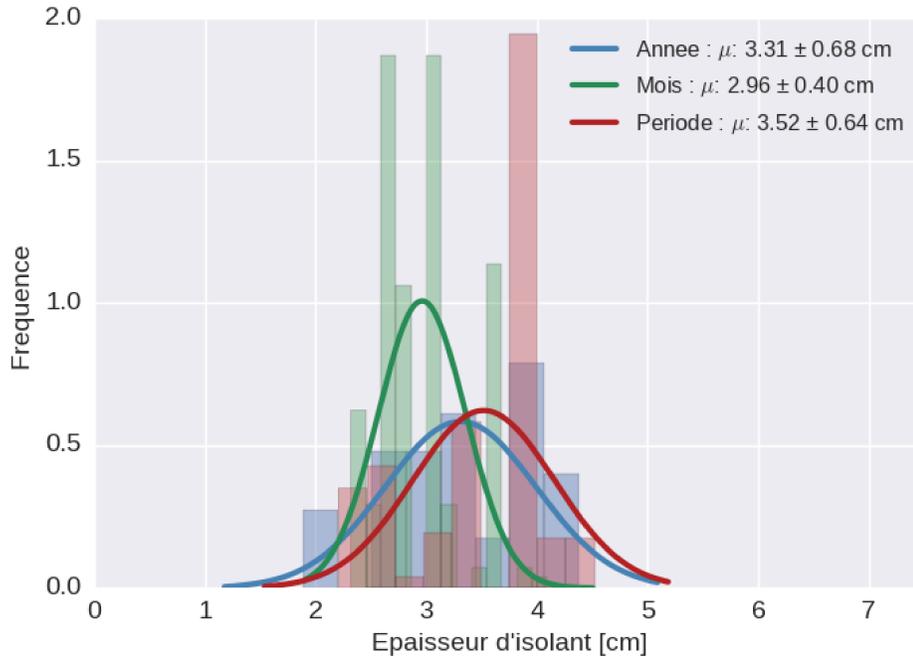


Figure III.13 – Résultat de l’algorithme ABCPMC appliqué à l’écart de températures pour différentes durées de simulation

Pour chacune de ces périodes, notre méthodologie a été appliquée. L’indicateur est ici l’écart de la température ($C_{T_{int}}$) dans la chambre parentale. Il est calculé sur la période d’absence hivernale soit du 8 au 12 février inclus (5 jours). Nous avons choisi d’accepter 200 valeurs à chaque itération avec le critère ϵ suivant la décroissance linéaire exprimée dans le tableau III.1. Le résultat de l’algorithme ABCPMC pour chaque cas est présenté sur la figure III.13.

- Cas 1 : En bleu est représenté le résultat lorsque la simulation est effectuée sur une année complète. La valeur retrouvée est ici de 3,31 cm avec une incertitude de 0,68 cm. Ce cas là a nécessité 8h30 pour environ 48 000 simulations, ce qui est assez conséquent.
- Cas 2 : En vert, il s’agit des résultats lorsque l’on simule à partir de 30 jours avant la période d’absence jusqu’à la fin de celle-ci. La valeur retrouvée est de 2,96 cm avec une incertitude de 0,4 cm avec un temps de calcul de 50 minutes pour 8800 simulations.
- Cas 3 : En rouge, on trouve finalement le cas où la simulation n’est effectuée que sur la période d’absence (du 8 au 12 février inclus). La valeur retrouvée par l’ABCPMC est 3,51 cm avec une incertitude de 0,64 cm. Le temps de calcul est cette fois très rapide, 20 minutes, pour 5470 simulations.

De par ces résultats, la période du cas 2 est la plus appropriée avec le résultat avec la plus faible incertitude, et une moyenne très proche du résultat recherché (3 cm). On observe bien

l'inadéquation d'une simulation sur une année complète : c'est extrêmement chronophage et le résultat n'est pas plus précis (et même moins) que sur une simulation sur une plus courte durée. Le cas 3 est également à éviter car le résultat est trop éloigné de la valeur recherchée. Le cas 2 est donc le bon compromis entre précision et temps de simulation. Les essais seront donc effectués sur cette durée par la suite.

4.2 Durée d'absence optimale

Précédemment, une période d'absence de 5 jours a été identifiée sur les mesures de la maison considérée. Les essais ont donc commencé en étudiant cette durée d'absence. Cependant, l'étude portant sur deux modèles, il est possible de l'augmenter ou la réduire virtuellement, afin de voir comment la durée d'absence influe sur le résultat de la méthodologie.

L'étude suivante a donc été menée :

- Les deux modèles précédents sont étudiés : l'un sans défaut et l'autre avec une épaisseur d'isolant de 3 cm au lieu des 18 initiaux. On vise donc à retrouver une épaisseur de 3 cm par l'application de cette méthode.
- La période de simulation commence 30 jours avant le début de la période d'absence et finit à la fin de celle-ci.
- Différentes durées d'absence sont simulées : de 1 jour à 30 jours par pas de 1 jour.
- La méthode étant stochastique, elle est appliquée 10 fois pour chaque durée d'absence ce qui permet d'avoir une tendance du résultat pour chaque durée.

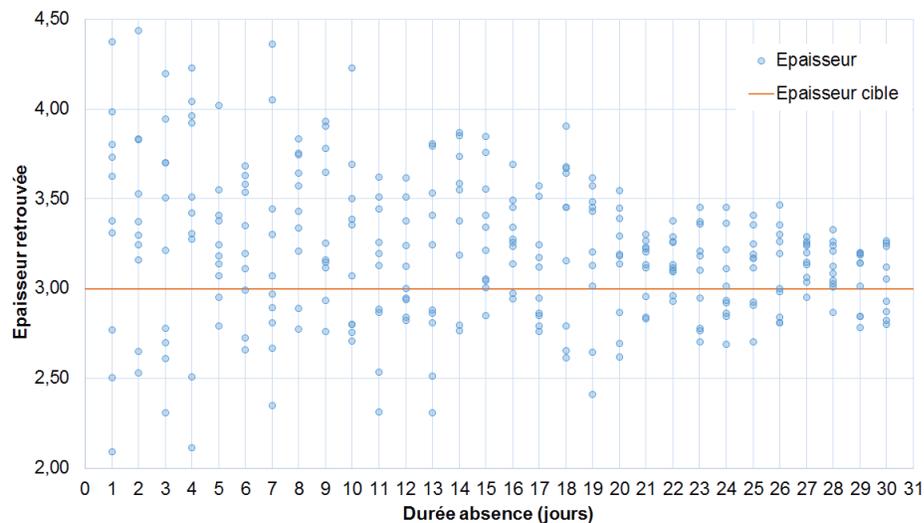


Figure III.14 – Résultats de l'algorithme ABCPMC appliqué à l'écart de températures selon différentes durées d'absence

La figure III.14 présente les résultats de cette étude. Les valeurs retrouvées sont représentées par les ronds bleus. On peut donc apercevoir ici une tendance : plus la période d'absence est longue, plus le résultat tend à être précis.

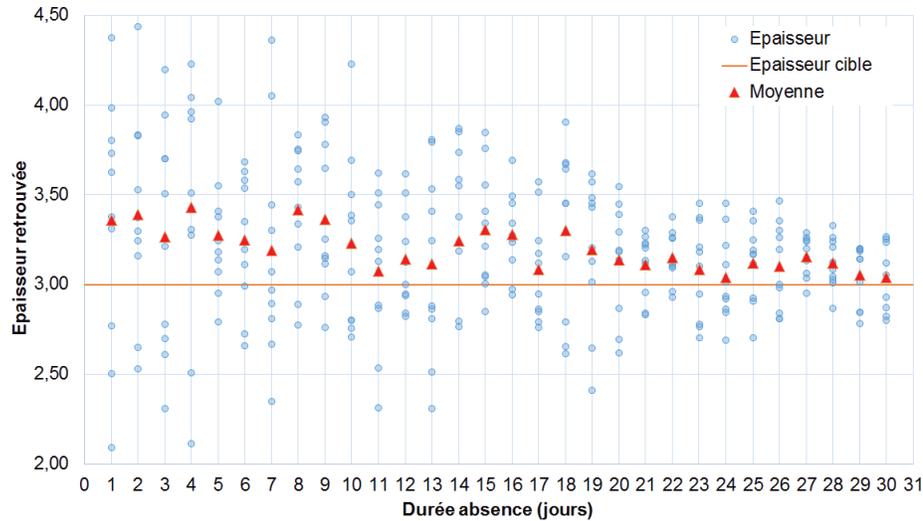


Figure III.15 – Moyennes pour chaque durée d'absence des résultats de l'algorithme ABCPMC appliqué à l'écart de températures

Pour chaque durée d'absence, la moyenne des résultats est calculée. Elle est représentée par les triangles rouges sur la figure III.15. On constate que celle-ci se rapproche également de plus en plus de l'objectif recherché : les 3 cm.

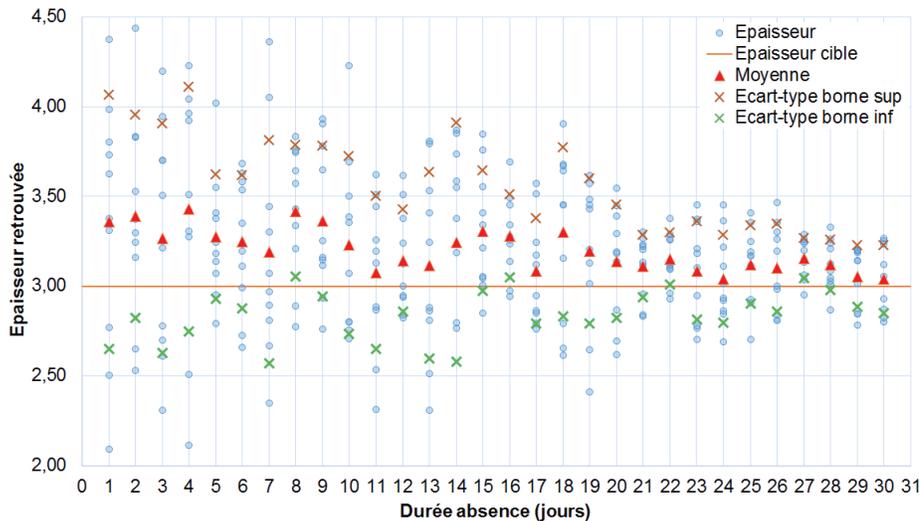


Figure III.16 – Ecart-types (bornes supérieure et inférieure) pour chaque durée d'absence des résultats de l'algorithme ABCPMC appliqué à l'écart de températures

Afin de vérifier cela, l'écart-type est également calculé pour chaque durée d'absence. Sa borne supérieure est représentée par les croix oranges et sa borne inférieure par les croix vertes sur la figure III.16. La tendance semble se confirmer.

En traçant les courbes de tendance des bornes supérieure et inférieure (droites noires sur la figure III.17), on visualise d'autant plus la tendance.

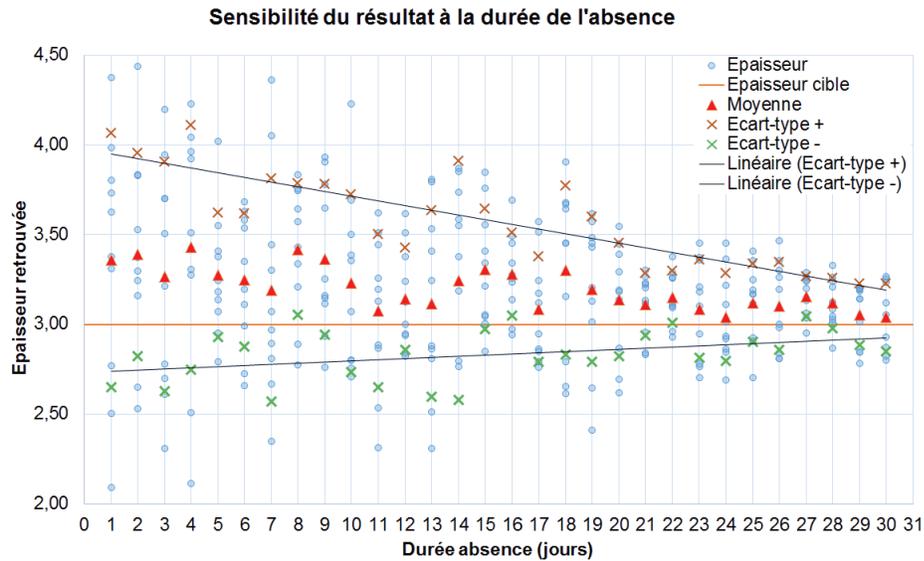


Figure III.17 – Tendence globale des résultats de l'algorithme ABCPMC appliqué à l'écart de températures

La première chose remarquable est que plus la période d'absence augmente, plus l'écart-type diminue. En d'autres termes, plus on a des données, plus la méthode est capable de retrouver un résultat avec une faible incertitude, donc un résultat fiable. Nous pouvions nous douter de cette conclusion car cela semble logique. Par cette étude, nous avons toutefois testé et quantifié cette hypothèse sur cet exemple. Ensuite, on constate que la moyenne se rapproche de plus en plus des 3 cm recherchés. Enfin, les résultats rencontrés pour de courtes absences ont une plus grande incertitude mais celle-ci reste relativement faible, environ 0,6 cm.

Pour ce cas particulier, nous pouvons conclure qu'une durée d'absence plus grande mène à des résultats plus précis et plus stables.

4.3 Conclusion sur l'échelle temporelle

Dans cette partie, l'aspect temporel a été étudié afin d'évaluer d'une part la pertinence d'effectuer les simulations sur une année complète (III.4.1) et d'autre part l'impact de la durée de la période d'absence sur le résultat (III.4.2).

Dans la première partie (III.4.1), il a été montré qu'une simulation annuelle n'apportait pas de résultat plus fiable qu'une sur une période plus courte. La période de simulation retenue est celle commençant 30 jours avant la période d'absence identifiée et finissant à la

fin de cette dernière. Il doit donc exister un optimum pour la durée de la simulation à valider par l'application de la méthodologie sur d'autres cas.

Dans la seconde partie (III.4.2), un fait important a été mis en évidence : l'incertitude sur le résultat, bien que faible sur une courte période d'absence, devient minime pour une période d'absence longue (20 à 30 jours). Cela fait donc émerger ce qui semble être un ordre de grandeur pour une période courte et longue pour l'application de notre méthodologie.

5 Etude de l'aspect spatial : à quelle échelle arrive-t-on à identifier un défaut ? △

Une fois le temps de simulation et la période d'absence nécessaires pour avoir des résultats satisfaisants définis, il reste à connaître l'échelle à laquelle un défaut peut être identifié : un défaut global, c'est-à-dire sur toutes les parois de la maison, puis un défaut local soit à l'échelle d'une pièce soit à l'échelle d'une paroi. En d'autres termes, nous cherchons à savoir jusqu'à quelle échelle d'espace le défaut est identifiable grâce à la méthodologie développée. Cette partie va nous permettre de répondre à cette question.

5.1 Défaut minimal détectable par la méthode

Dans cette partie, nous nous intéressons à l'intensité du défaut engendré. Jusqu'à présent, nous avons étudié le cas d'un défaut de 15 cm, soit d'une épaisseur d'isolant de 3 cm, ce qui est un défaut caricatural utilisé pour tester la méthodologie. Ici, nous allons étudier le résultat de l'algorithme bayésien ABCPMC appliqué toujours à l'écart de température pour une épaisseur variant entre 1 et 17 cm afin d'éprouver la méthodologie lorsque l'on se rapproche d'un défaut crédible pouvant correspondre à des défauts réels.

La figure III.18 présente les différents résultats obtenus représentés par les ronds bleus. La tendance des résultats, symbolisée par la droite noire, semble assez linéaire avec un coefficient r^2 égal à 0.97. La courbe cible, que nous cherchons à atteindre, est une courbe linéaire $y = x$ tracée en orange. En effet, lorsque nous insérons un défaut dans un modèle, nous introduisons une nouvelle épaisseur, et recherchons donc celle-ci lors de l'application de l'algorithme bayésien.

On peut constater que plus l'épaisseur augmente (et donc que le défaut diminue), plus la courbe noire dévie de la courbe cible orange. En effet, on peut observer qu'à partir de 10cm, le résultat devient plus incertain, variant de 1 à 2 cm autour de l'épaisseur cible considérée.

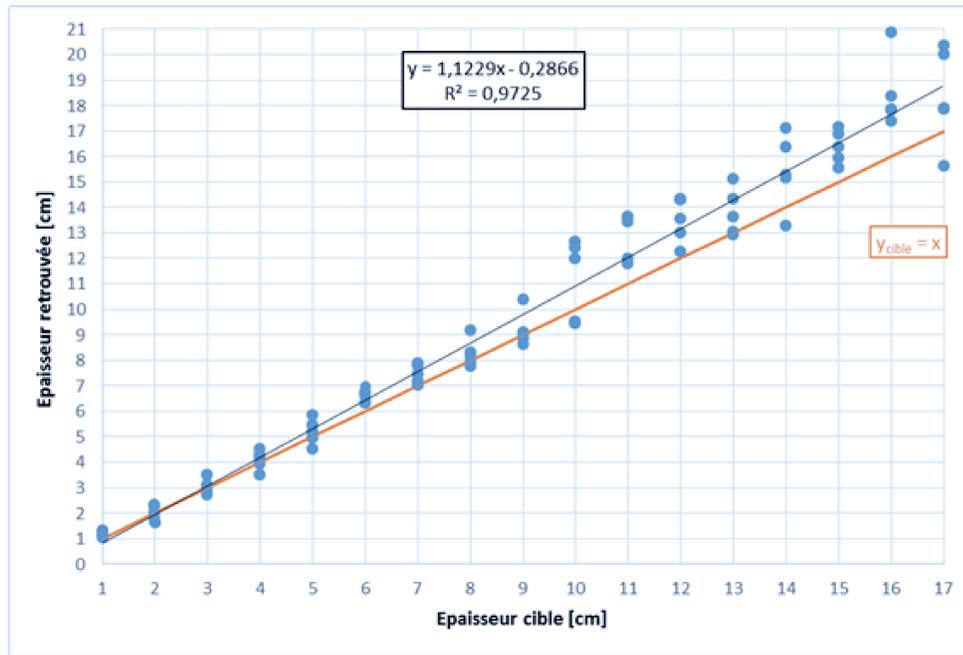


Figure III.18 – Evolution du résultat selon le degré de défaut inséré.

Le défaut devient donc de plus en plus difficile à détecter et la méthode est de moins en moins discriminante.

5.2 Rôle de la durée de l'absence sur la détectabilité du défaut

Fort du constat qu'à partir de 10 cm, et pour une durée d'absence de 5 jours, le défaut devenait de moins en moins détectable (5.1), nous voulions savoir ce qu'il se passerait si la durée d'absence augmentait. Dans la partie 4.2, nous avons vu que, pour une épaisseur établie à 3 cm, plus la durée d'absence augmente plus l'incertitude sur le résultat devient faible.

5.2.1 Evolution de l'épaisseur retrouvée selon la durée d'absence et l'épaisseur cible

De ces deux résultats, nous avons donc cherché à établir le même constat mais pour des épaisseurs se rapprochant de l'épaisseur initiale, et donc pour un défaut plus minime. Nous avons ainsi, pour chaque épaisseur comprise entre 10 et 17 cm, augmenté la durée de l'absence, de 5 à 30 jours.

La figure III.19 présente les résultats pour une épaisseur entre 10 et 13 cm. On observe clairement que plus l'épaisseur est faible (donc le défaut important) et plus la durée d'absence est grande, plus le résultat a tendance à être correct.

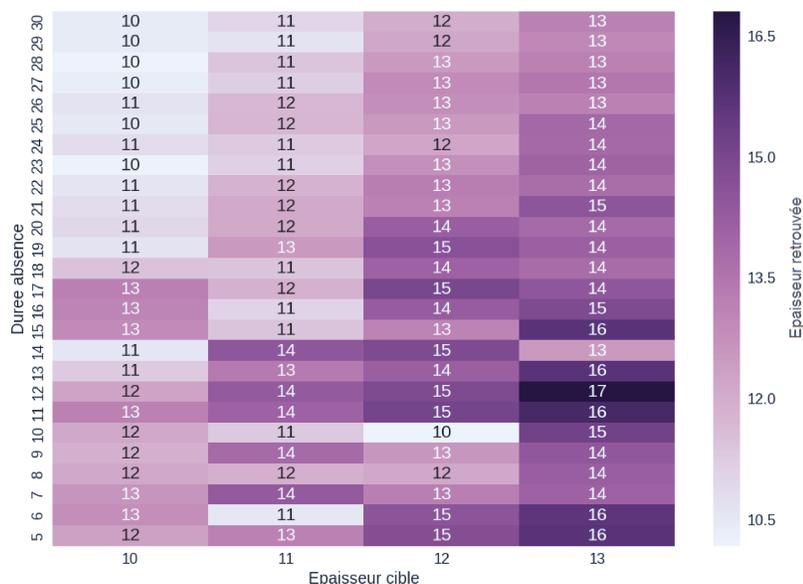


Figure III.19 – Epaisseurs entre 10 et 13 cm : évolution du résultat selon la durée d'absence.

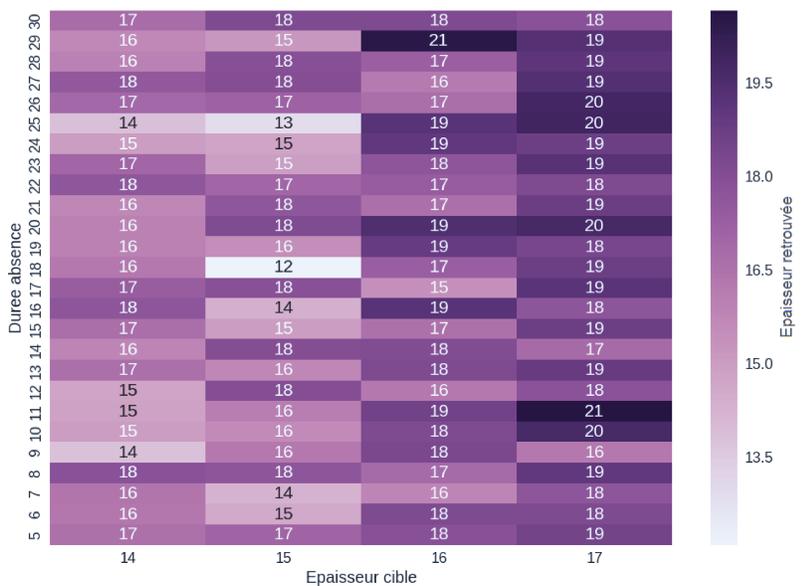


Figure III.20 – Epaisseurs entre 14 et 17 cm : évolution du résultat selon la durée d'absence.

La figure III.20 présente les résultats pour une épaisseur entre 14 et 17 cm. Dans ce cas là, le résultat a tendance à être plus proche de l'épaisseur initiale, 18 cm, que de l'épaisseur cible.

Enfin, la figure III.21 présente les résultats dans leur ensemble, soit pour une épaisseur entre 10 et 17 cm, dans le cas particulier de la maison POS-A. Sur cette figure, on remarque clairement une coupure entre 13 et 14 cm : à 13 cm le résultat est proche de l'épaisseur cible (13 cm) tandis qu'à 14 cm, il se rapproche de l'épaisseur initiale (18 cm).

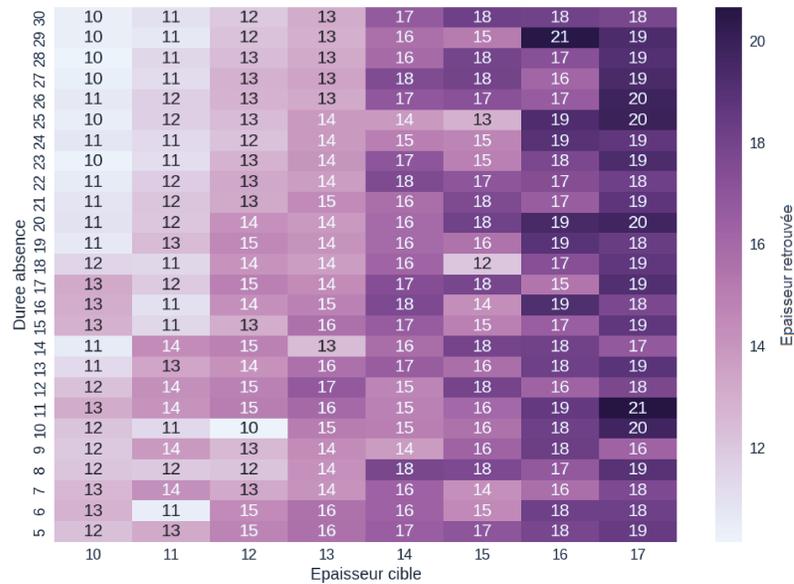


Figure III.21 – Epaisseurs entre 10 et 17 cm : évolution du résultat selon la durée d'absence.

Les figures III.19, III.20 et III.21 sont cependant à prendre avec du recul. En effet, elles présentent seulement l'épaisseur retrouvée associée à une échelle de couleurs : plus l'épaisseur est grande, plus la couleur est foncée. Certes, elles permettent en un regard de comprendre l'évolution de l'épaisseur retrouvée selon la distance mais pour analyser la fiabilité du résultat, il est important d'étudier plutôt l'écart à l'épaisseur cible.

5.2.2 Evolution de l'écart absolu à l'épaisseur cible selon la durée d'absence

Nous présentons les mêmes résultats mais cette fois mis à l'échelle de l'écart absolu à l'épaisseur cible sur les figures III.22, III.23 et III.24.

La figure III.22 présente l'évolution de l'écart absolu pour des épaisseurs cibles entre 10 et 13 cm selon la durée de l'absence. On peut voir une tendance se dessiner aux alentours des 15 jours d'absence, avec un écart absolu inférieur à 1 cm.



Figure III.22 - Epaisseurs entre 10 et 13 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.

La figure III.23 représente l'évolution de l'écart absolu mais cette fois entre 14 et 17 cm. Le résultat est ici plus mitigé : pour de courtes durées, on a quelques fois un écart faible, inférieur à 1 cm, et pour de longues durées, des écarts importants.

Enfin, la figure III.24 concatène les deux figures précédentes. On peut y observer une démarcation nette en forme d'arc de cercle pour des épaisseurs cibles entre 10 et 13 cm et pour une durée d'absence de 18 jours à plus. Dans ce quart de cercle, l'écart absolu est faible, entre 1 et 0,01 cm. Il semble donc exister ce que l'on pourrait appeler un front de discriminabilité dans l'espace « Durée absence - Epaisseur cible » permettant d'identifier clairement l'optimum de fonctionnement de la méthodologie. Pour le reste de la figure, il semble avoir un rapprochement vers l'épaisseur initiale, 18 cm, ou tout du moins, une oscillation autour de cette épaisseur à ± 1 cm.

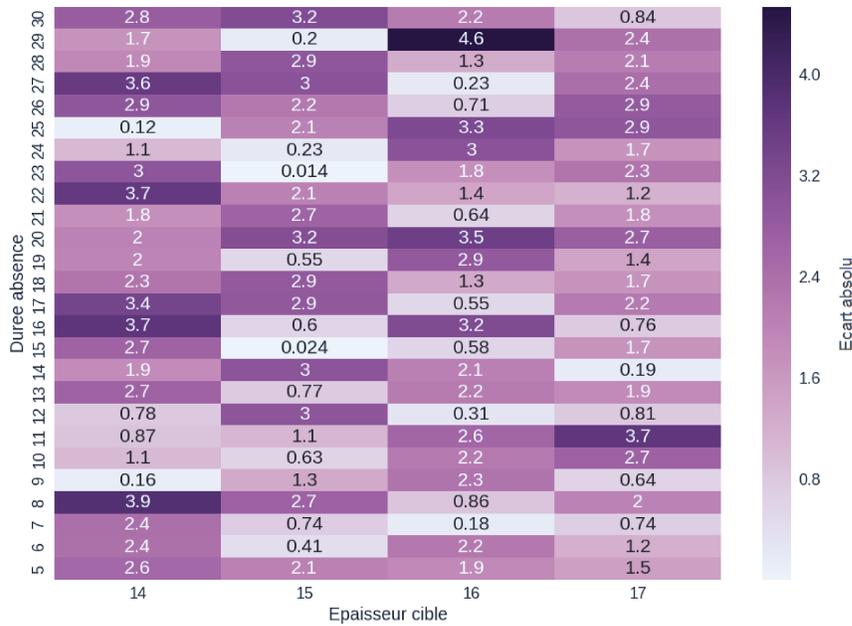


Figure III.23 - Épaisseurs entre 14 et 17 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.

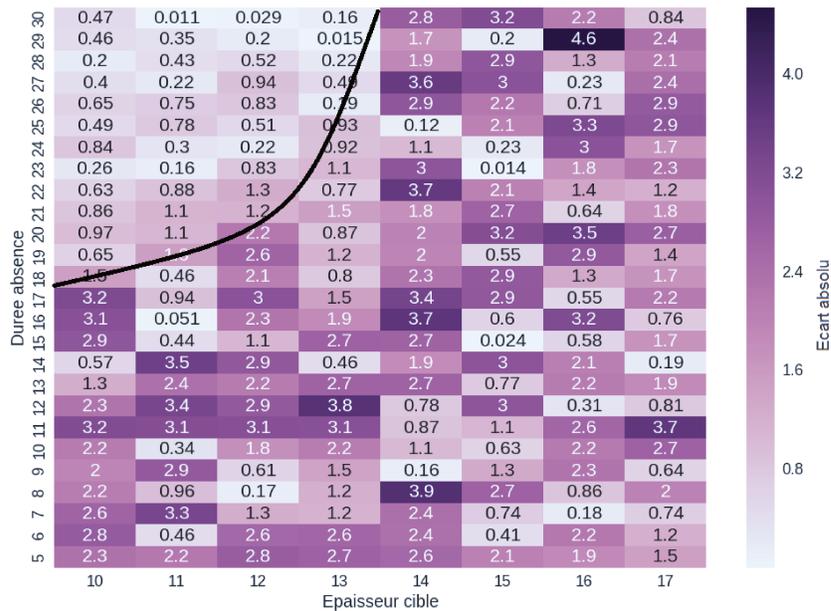


Figure III.24 - Épaisseurs entre 10 et 17 cm : évolution de l'écart absolu à l'épaisseur cible selon la durée d'absence.

5.2.3 Conclusion sur le rôle de la durée d'absence sur la détectabilité d'un défaut

Dans cette partie, nous avons vu que si l'on augmente la durée d'absence, nous arrivons à obtenir un résultat précis jusqu'à une épaisseur de 13 cm, soit un défaut de 5 cm sur l'épaisseur de l'isolant. Si le défaut diminue encore plus, le résultat tend vers l'épaisseur

initiale 18 cm. Cela s'explique par le fait qu'il s'agit d'une maison performante, sous un climat doux (Vaucluse, France) et donc qu'un écart de 5 cm sur son isolation n'impacte pas énormément la température intérieure.

5.3 Détection d'un défaut global

Dans cette partie, nous avons introduit le même défaut que dans la section précédente : une épaisseur d'isolant de 3cm au lieu des 18 initialement prévu, soit un défaut d'isolation de 15cm sur l'ensemble des murs extérieurs de la maison.

5.3.1 Résultat de l'analyse de Morris dans le cas d'un défaut global

La figure III.25 présente les résultats de l'analyse de Morris. Nous avons effectué cette analyse sur les 131 facteurs précédemment étudiés (3). L'indicateur est le $C_v(\text{RMSE})$ calculé sur la température intérieure. Notre but ici était de voir si notre défaut en termes d'isolation était repérable lors de l'étude de n'importe quelle zone de la maison. Nous avons donc calculé le $C_v(\text{RMSE})$ (II.3) sur la température intérieure de chaque zone. Nous pouvons voir sur la figure III.25 que les facteurs 1, 4 et 10 ressortent sur toutes les pièces considérées. Ceux-ci correspondent à notre isolant et respectivement à son épaisseur, sa conductivité et sa densité. Cela signifie donc que la méthode de Morris identifie l'isolant altéré comme un des principaux facteurs influençant l'écart de températures.

Nous voulons ici seulement vérifier si ce défaut est bien repérable par l'analyse des différentes zones du modèle : la chambre parentale (que nous avons étudiée jusqu'à présent), le salon/cuisine, le bureau et les chambres du rez-de-chaussée. La figure III.25 présente le résultat de l'analyse de Morris pour chacune de ces zones : on peut remarquer que l'épaisseur et la conductivité de l'isolant ressort dans chaque zone, ce qui indique que ces deux facteurs sont ceux ayant le plus d'effet sur l'écart de températures.

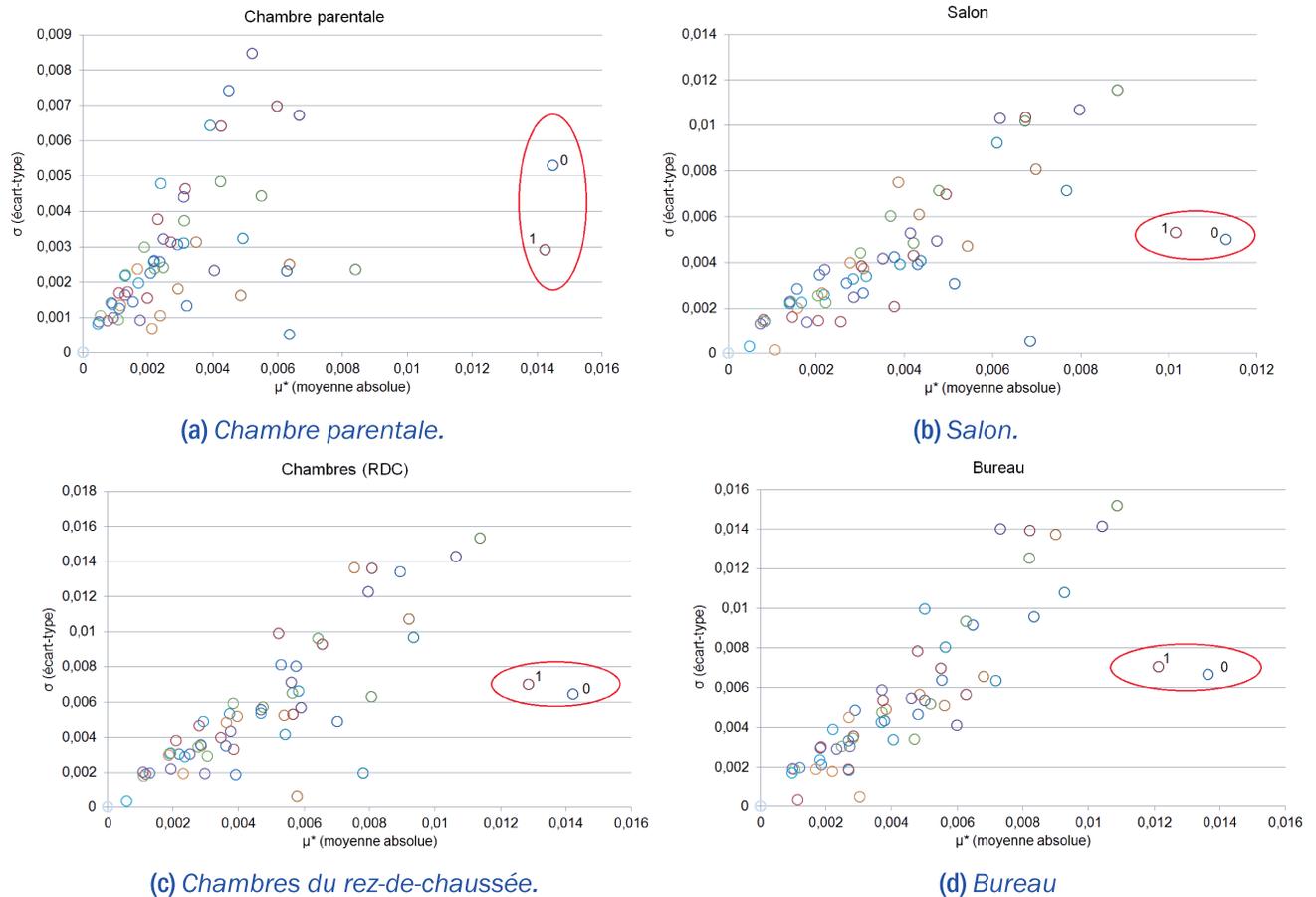


Figure III.25 – Analyse de Morris : facteurs les plus influents lors de la recherche d'un défaut global.

5.3.2 Résultat de l'algorithme ABCPMC dans le cas d'un défaut global

Le résultat a déjà été présenté sur la figure III.6 lorsque nous avons étudié les deux indicateurs possibles. Nous avons trouvé une épaisseur de 3,2 cm avec une incertitude de 0,8 cm. Dans ce cas particulier, de conclure que la méthodologie fonctionne.

5.4 Détection d'un défaut local à l'échelle d'une pièce

5.4.1 Mise en place de l'étude

Nous allons ici essayer d'identifier un défaut à l'échelle d'une pièce. Nous allons introduire le défaut (une épaisseur d'isolant de 3cm) sur toutes les parois de la chambre parentale tandis que les autres parois gardent une épaisseur d'isolant de 18 cm. Nous différencions les matériaux utilisés dans la chambre de ceux utilisés dans le reste du bâtiment mais tous seront soumis à l'étude.

5.4.2 Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une pièce

La figure III.26 montre le résultat de l'analyse de Morris sur l'écart de la température intérieure dans le cas où le défaut est introduit seulement sur la chambre parentale. Le tableau III.4 permet de légender cette figure.

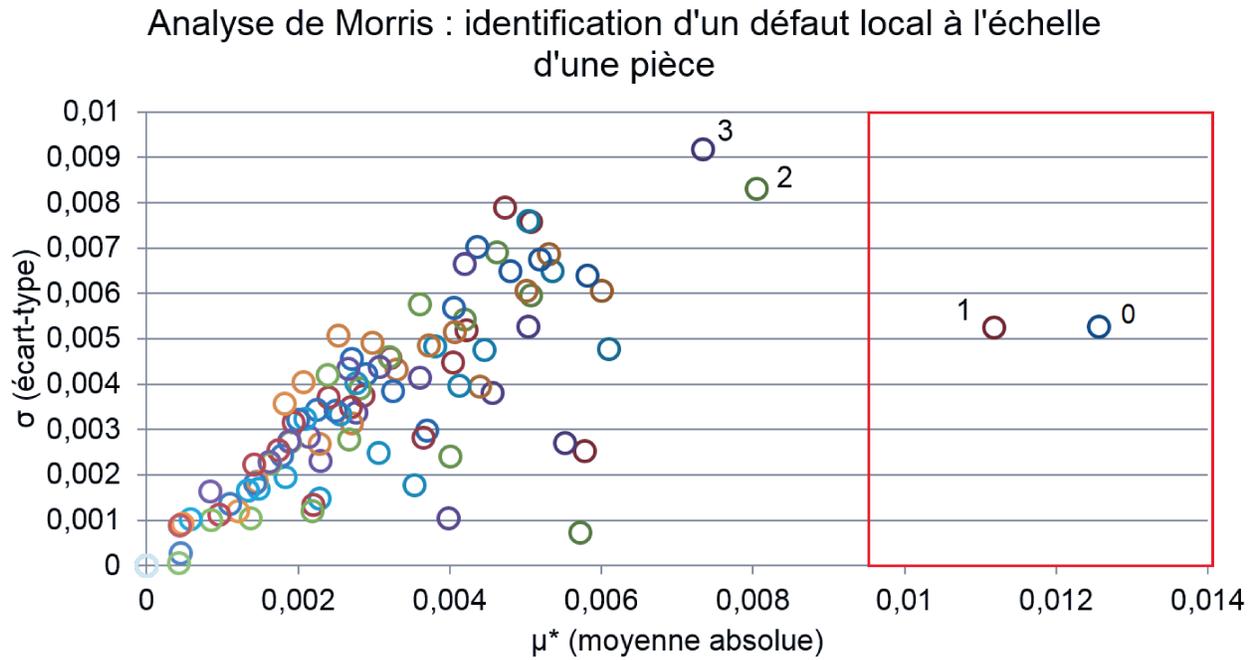


Figure III.26 – Altération de l'isolation de la chambre parentale : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver

Index	Facteur
0	Laine minérale chambre - Epaisseur
1	Laine minérale chambre - Conductivité
2	Laine minérale chambre - Densité
3	Entrevous en polystyrène chambre - Conductivité

Table III.4 – Légende figure III.26.

On peut remarquer que seuls des facteurs liés à la chambre ressortent, et l'isolant d'autant plus : la conductivité (1) et l'épaisseur (0) de l'isolant situé dans la chambre parentale sont les facteurs ayant le plus d'effet sur l'indicateur de températures.

L'analyse de Morris semble donc faire ressortir le défaut que nous avons inséré. Il reste à vérifier que l'algorithme ABCPMC retrouve bien une loi de dispersion centrée autour du défaut inséré, c'est-à-dire 3 cm.

5.4.3 Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une pièce

Suite aux résultats de l'analyse de Morris sur la chambre parentale, l'algorithme ABCPMC est appliqué sur l'épaisseur de l'isolant de cette pièce et sur sa conductivité. Nous gardons comme critères une décroissance linéaire sur 10 itérations de 5°C à $0,2^{\circ}\text{C}$. Le nombre de valeurs à accepter pour chaque itération est de 5000.

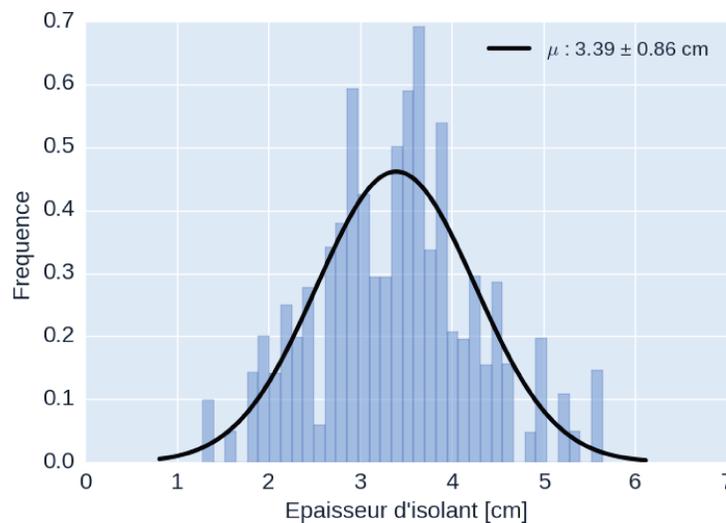


Figure III.27 – Altération de l'isolation de la chambre parentale : Résultat de l'algorithme ABCPMC pour l'épaisseur de l'isolant

La figure III.27 présente le résultat sur l'épaisseur de l'isolant. On observe une gaussienne centrée sur 3,39 cm avec une incertitude de 0,8 cm. Par rapport à l'épaisseur recherchée, 3 cm, on a donc une erreur d'environ 10%, moins précis que pour le défaut global où nous avons une erreur d'environ 6%.

5.5 Détection d'un défaut local à l'échelle d'une paroi

5.5.1 Mise en place de l'étude

Nous allons ici essayer d'identifier un défaut à l'échelle de la paroi. Pour cela, 3 analyses de Morris sont effectuées, permettant ainsi d'isoler soit la paroi Sud soit la paroi Nord soit la paroi Est de la chambre parentale : une paroi est isolée par analyse, les autres restent à l'épaisseur d'isolant initiale de 18 cm.

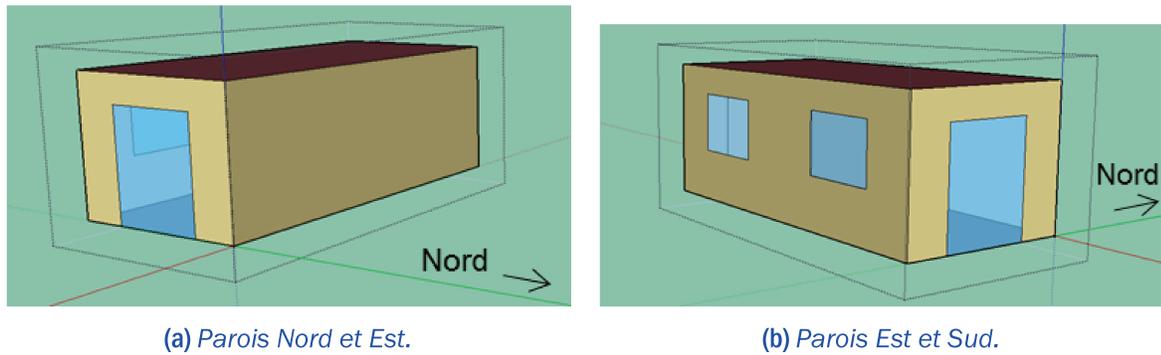


Figure III.28 - Maison POS-A - Parois

5.5.2 Résultat de l'analyse de Morris dans le cas d'un défaut introduit seulement sur une paroi

Les figures III.29, III.30 et III.31 présentent respectivement les résultats lorsque le défaut est appliqué à la paroi Est, la paroi Sud et la paroi Nord. Ces trois parois, outre leur exposition, ont des caractéristiques différentes : la paroi Nord a une surface de 20 m^2 sans fenêtre, la paroi Sud a la même surface que la paroi Nord mais avec deux fenêtres d'une surface totale de 4 m^2 et la paroi Est a une surface de 10 m^2 dont une baie vitrée de 5 m^2 . On risque donc d'avoir des résultats variables selon la paroi étudiée. La paroi Ouest n'est pas étudiée car il s'agit d'une paroi intérieure, liant la chambre parentale à l'espace séjour.

Sur chacun des graphiques, la légende « 0 » correspond à l'effet de l'épaisseur de la laine minérale de la paroi isolée, « 1 » à celui de l'épaisseur de la laine minérale sur les autres parois et « 2 » à celui de l'épaisseur de la laine minérale sur le reste du bâtiment.

La figure III.29 présente le résultat sur la paroi Est de la chambre parentale. On peut remarquer que l'épaisseur d'isolant de cette paroi (0) est loin d'avoir un effet important sur l'écart de températures. Ce facteur est en effet situé en bas à gauche du graphique, entrant donc dans la zone des effets négligeables. D'autres facteurs ont plus d'influence tels que l'épaisseur de l'isolant sur les autres parois de la chambre (1). L'épaisseur du reste du bâtiment (2) a un effet plus important mais reste léger par rapport à celle des autres parois de la chambre parentale. Cela s'explique par le fait que cette paroi n'a qu'une faible surface de laine minérale. Ainsi, un défaut d'isolant sur cette paroi impacte peu l'écart de température.

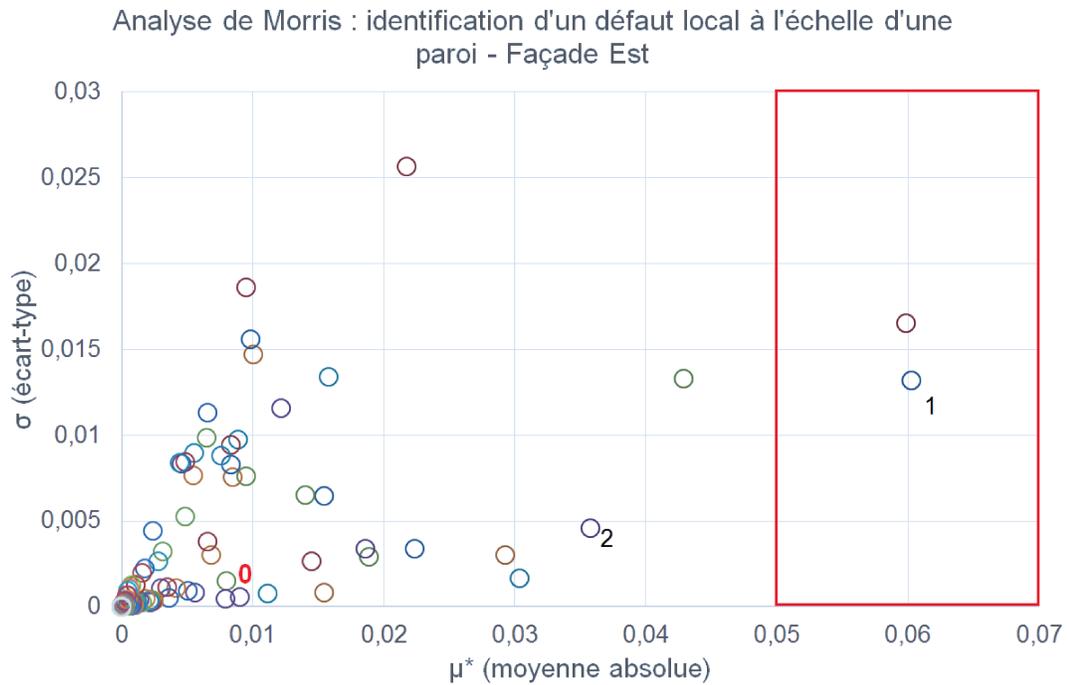


Figure III.29 – Altération de l'isolation de la façade est : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver

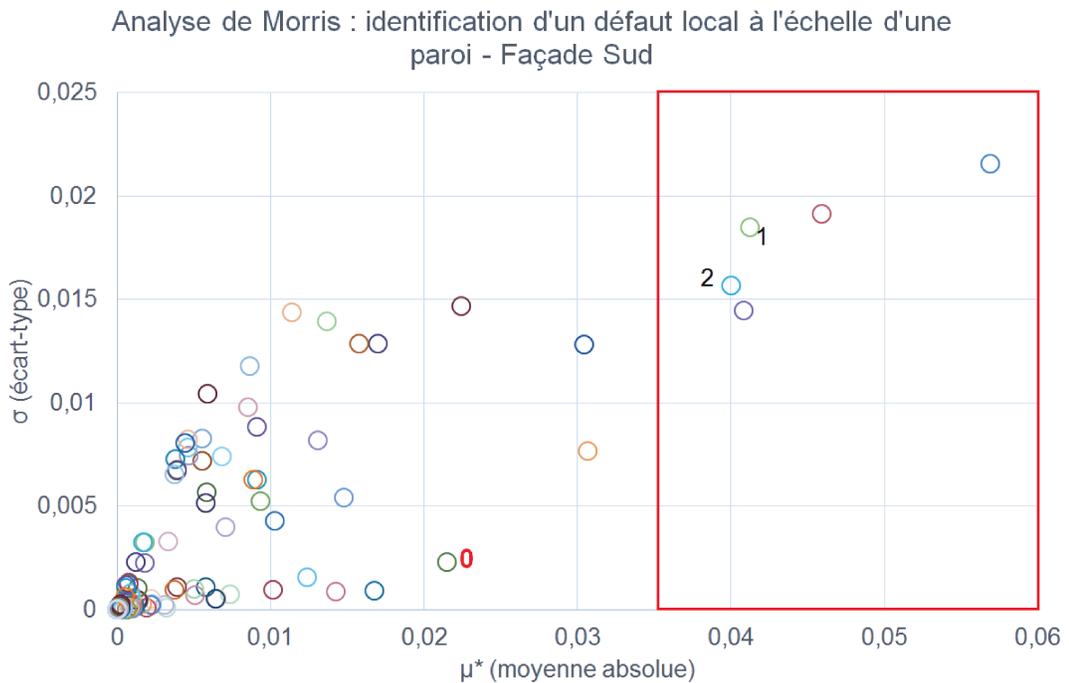


Figure III.30 – Altération de l'isolation de la façade sud : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver

La figure III.30 présente le résultat de l'analyse de Morris sur la façade Sud de la chambre parentale. Comme pour la figure précédente, l'épaisseur d'isolant de la paroi considérée se

situé en bas à gauche du graphique, indiquant ici que son effet est négligeable par rapport aux autres facteurs. On peut également constater que l'isolant des autres parois de la chambre (1) et celui du reste du bâtiment (2) font partie des facteurs à effet important. Ce résultat peut s'expliquer par les apports solaires importants reçus par cette paroi. En effet, la maison POS-A se situe dans un climat doux donc elle reçoit de forts apports solaires, même sur la période hivernale. Il est alors probable que, comme sur la période d'absence estivale, les matériaux à inertie ont plus d'influence dans ce cas-ci.

Enfin, la figure III.31 présente enfin le résultat de l'analyse de Morris sur la paroi nord de la chambre parentale. Contrairement aux deux figures précédentes, on peut constater que l'épaisseur de l'isolant de la paroi considérée (0) est le facteur dont l'effet est le plus important par rapport aux autres, bien que l'isolant des autres parois (1) et celui du reste du bâtiment (2) sont également influents.

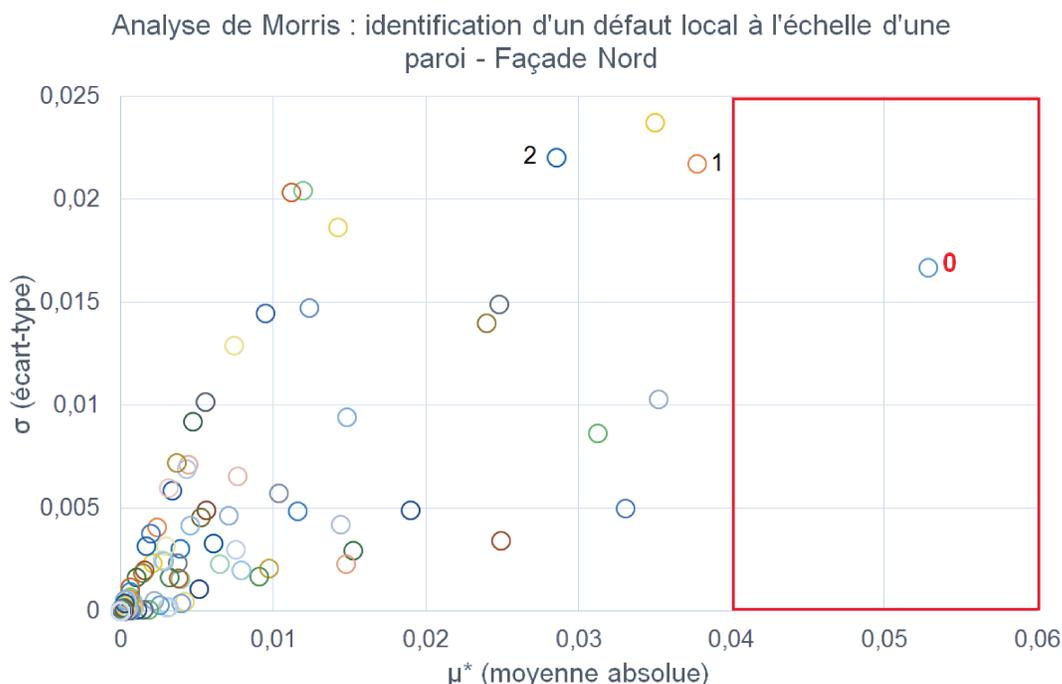


Figure III.31 – Altération de l'isolation de la façade nord : résultat de l'analyse de Morris sur l'écart de températures pour l'échantillon hiver

A l'étude de ces trois figures, il apparaît que la méthode identifie bien l'isolant de la paroi considérée seulement pour la paroi nord. Cela pourrait être dû à plusieurs facteurs :

- la façade Nord est la seule sans fenêtre : la surface d'isolant est donc la plus grande sur cette paroi (20 m²). Une réduction de son épaisseur a donc plus d'effet que sur les autres façades.

- Les apports solaires dus à la présence de fenêtre(s) sur les façades Est et Sud compensent potentiellement la différence de températures, ce qui rend ainsi difficile à la méthode de Morris d'identifier le facteur influent.

L'analyse de Morris a donc réussi à détecter l'isolant de la paroi nord comme très influent. Cela nous pousse à penser que l'analyse de Morris doit être effectuée dans des conditions favorables lorsque le défaut recherché se situe seulement sur une paroi : la surface de celle-ci doit être suffisamment importante pour que le défaut lui soit attribué plutôt qu'aux caractéristiques des autres parois, et, en plus, si les apports solaires sont importants, comme sur la façade sud, les matériaux à inertie stockent suffisamment de chaleur pour que cela perturbe l'analyse.

5.5.3 Résultat de l'algorithme ABCPMC dans le cas d'un défaut introduit seulement sur une paroi

Nous allons malgré tout appliquer l'algorithme ABCPMC sur l'épaisseur de laine minérale de la paroi nord. En effet, comme l'analyse de Morris a réussi à l'identifier comme facteur influent, nous pouvons continuer dans notre méthodologie et appliquer l'algorithme ABCPMC afin de voir si nous retrouvons les 3 cm que nous avons insérés. Nous l'appliquons donc sur l'épaisseur de laine minérale de cette paroi. Nous gardons toujours comme critères une décroissance linéaire sur 10 itérations de 5°C à $0,2^{\circ}\text{C}$. Le nombre de valeurs à accepter pour chaque itération est de 5000.

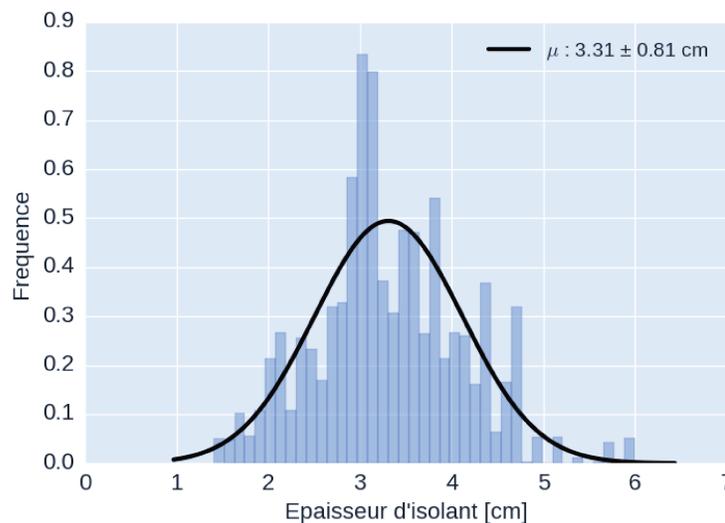


Figure III.32 – Altération de l'isolation de la façade nord : Résultat de l'algorithme ABCPMC pour l'épaisseur de l'isolant

La figure III.32 présente le résultat sur l'épaisseur de la laine minérale sur la paroi Nord. On observe une gaussienne centrée sur 3,31 cm avec une incertitude de 0,8 cm. Par rapport

à l'épaisseur recherchée, 3 cm, on a donc une erreur d'environ 10%. Le résultat est, dans ce cas satisfaisant, mais nous ne devons pas oublier que la méthodologie n'a pu conclure que pour la façade Nord tandis que pour les façades Est et Sud, l'analyse de Morris n'est pas suffisante pour détecter le défaut inséré.

Ces résultats font donc apparaître une dualité dans notre méthodologie. En effet, nous pouvons nous demander si le problème vient de la méthodologie en soi qui ne serait pas assez discriminante, et n'arriverait donc pas à identifier le défaut, ou bien s'il vient du défaut qui ne serait pas assez discriminable, et donc non détectable par l'analyse.

5.6 Conclusion : quelles limites présente notre méthode en terme d'échelle spatiale ?

Nous avons étudié dans cette partie sur l'échelle spatiale plusieurs aspects.

Tout d'abord, nous avons identifié le défaut minimal détectable sur une période d'absence de cinq jours (5.1) : il s'agit d'un défaut de 8 cm sur l'épaisseur d'isolant.

Ensuite, comme nous avons vu à la partie 4.2 que plus la durée d'absence augmente, plus le résultat devient précis, nous avons regardé l'évolution du résultat pour une épaisseur variant de 10 à 17 cm et pour une durée d'absence variant de 5 à 30 jours (5.2). Nous avons ainsi pu observer que, pour une épaisseur entre 10 et 13 cm, l'augmentation de la durée d'absence permettrait effectivement d'obtenir un résultat plus précis, tandis que, pour une épaisseur entre 14 et 17 cm, le résultat ne varie pas énormément et reste proche de la valeur initiale du modèle soit une épaisseur de 18 cm.

Nous avons après cela essayé de voir jusqu'à quelle échelle le défaut était détectable en l'insérant soit sur tout le bâtiment (5.3), soit uniquement sur une pièce (5.4) soit uniquement sur une paroi (5.5).

Dans le premier cas, nous avons vu que le défaut semble détectable lors de l'étude de n'importe quelle pièce de la maison. Dans le deuxième, le défaut a été introduit seulement dans la chambre parentale, et cela fonctionne également. Enfin, pour l'étude à l'échelle de la paroi, nous avons testé la méthodologie sur les trois façades donnant sur l'extérieur de la chambre parentale : la façade Nord, Sud et Est. Sur la façade Nord, les résultats ont été concluants : l'analyse de Morris a permis d'identifier le défaut d'isolation sur la paroi et l'algorithme ABCPMC a retrouvé une valeur d'épaisseur de 3,31 cm. Pour les façades Est et Sud, le résultat a été tout autre avec de nombreux facteurs, et notamment les matériaux à inertie, qui perturbent l'analyse de Morris, ne permettant pas ainsi de détecter le défaut inséré comme facteur influent.

De manière plus générale, ces études montrent que l'algorithme bayésien fonctionne, que la méthode de Morris semble convenir mais elle montre toutefois ses limites lorsque le défaut inséré est un défaut très ciblé. Il faudrait donc peut-être combiner l'analyse de Morris avec une autre technique afin de consolider les résultats et ainsi d'avoir une réduction précise des facteurs influents permettant d'identifier un défaut. Cela nous amène donc à nous questionner sur la propriété de discrimination de la méthodologie et celle de discriminabilité du défaut pour son application.

6 Typologie de défauts détectables △

Nous nous sommes concentrés jusqu'à présent sur l'étude d'un défaut d'isolation, qui est un facteur important des déperditions thermiques dans un bâtiment. Or, celles-ci peuvent être causées par d'autres facteurs qu'il ne faut pas négliger. Dans cette partie nous allons ainsi nous atteler à l'étude d'autres facteurs potentiellement importants sur lesquels nous allons appliquer la méthodologie.

6.1 Défauts introduits

Nous allons ici insérer un seul défaut et effectuer pour chacun une analyse de Morris, puis appliquer l'algorithme ABCPMC.

Le tableau III.5 présente les différentes altérations que nous allons effectuer et tester. Nous nous intéressons principalement à d'autres matériaux et aux ponts thermiques. Nous avons exclu les scénarios de ventilation et de consommation électrique car ils sont à 0 lors des périodes d'absence.

Catégorie	Facteur altéré	Défaut inséré
Matériau	Béton d'argile - Epaisseur (20 cm)	10 cm
Matériau	Fenêtre - Facteur solaire (0,5)	0,1
Matériau	Fenêtre - Facteur solaire (0,5)	0,9
Matériau	Laine de verre - Epaisseur (16 cm)	3 cm
Matériau	Hourdis - Epaisseur (20 cm)	10 cm
Matériau	Chape béton - Epaisseur (6 cm)	2 cm
Infiltration	Ponts thermiques zone chauffée - Débit (0,01 m^3/s) ²	0,05 m^3/s

Table III.5 – Défauts testés.

6.2 Analyse de Morris pour chaque défaut

L'analyse de Morris n'a malheureusement pas porté ses fruits. En effet, aucun des défauts présentés ci-dessus n'a été identifié comme facteur influent sur l'écart de température. Seule la laine minérale, isolant de la façade extérieure, ressort systématiquement. Cependant, nous avons pu remarquer que sur certains défauts, l'analyse de Morris les faisait « remonter » par rapport à la première analyse effectuée : c'est que nous allons essayer d'expliquer dans les parties suivantes.

Pour chacun des graphiques de Morris ci-dessous, « 0 » représente l'épaisseur de la laine minérale, « 1 » sa conductivité et « 2 » la caractéristique que nous avons altérée dans le deuxième modèle.

6.2.1 Défauts repérables

Parmi ces facteurs se trouvent les facteurs solaires, les ponts thermiques et la laine de verre, que nous avons étudiée précédemment.

Les figures III.33 et III.34 présentent respectivement les résultats de l'analyse de Morris lorsque les « mesures » de température intérieure ont été générées à partir d'un modèle dont le facteur solaire des fenêtres a été altéré à 0,1 et à 0,9 alors que, dans le modèle initial, ce facteur est à 0,5. La figure III.35 représente quant à elle le résultat de l'analyse de Morris lorsque la mesure de température intérieure a été générée avec des ponts thermiques d'infiltrations à 0,05 m³/s.

Sur ces trois figures, les caractéristiques de la laine minérale ressortent, signifiant ainsi qu'une modification de la laine minérale semble influencer le plus l'écart de températures intérieures. Cependant, on peut remarquer que l'élément que nous avons modifié pour obtenir notre « mesure » de température a semble-t-il grimpé les échelons et est devenu plus influent par rapport à d'autres.

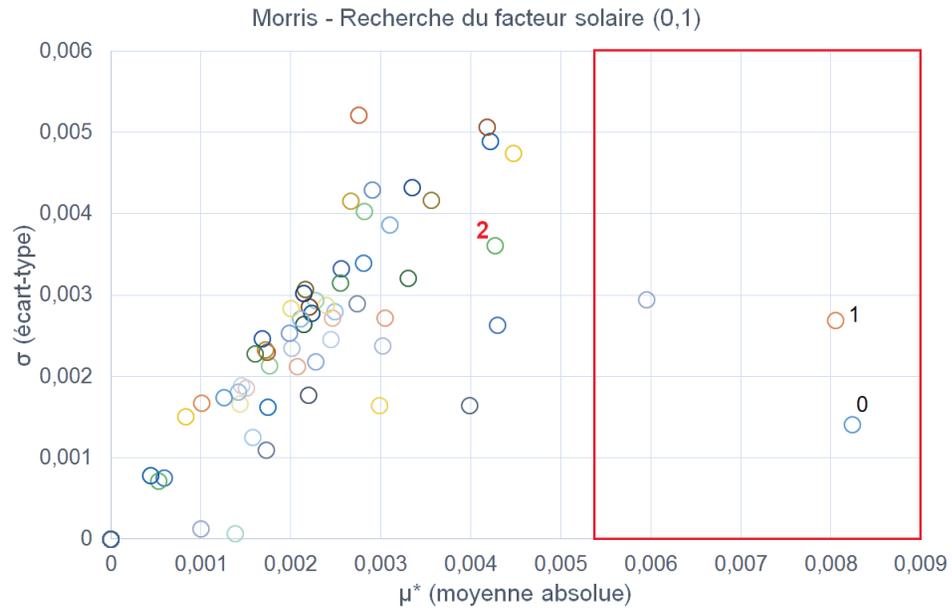


Figure III.33 - Ecart de températures pour l'échantillon hiver selon le défaut inséré.

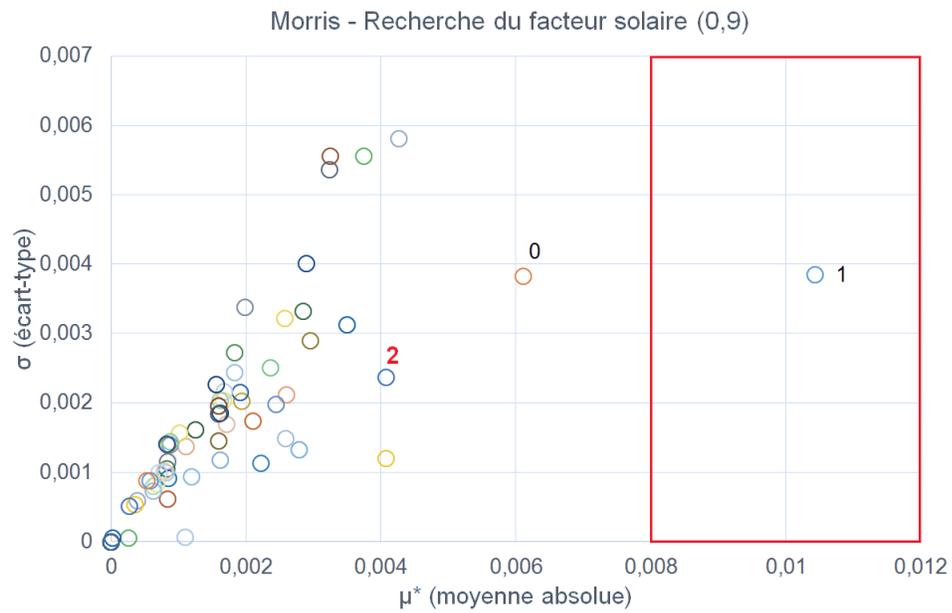


Figure III.34 - Ecart de températures pour l'échantillon hiver selon le défaut inséré.

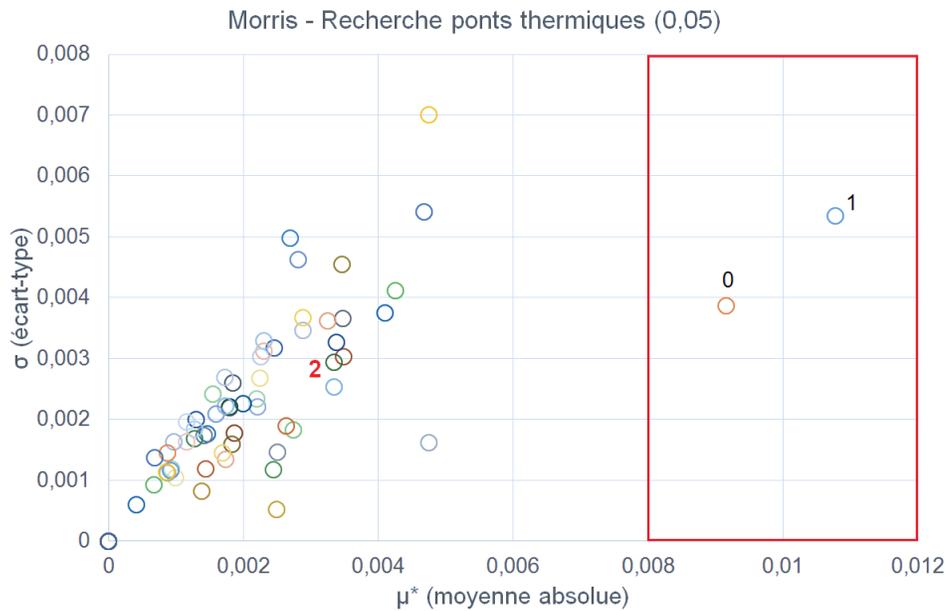


Figure III.35 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

Malgré cela, notre défaut ne serait pas notre première intuition pour l'application de l'algorithme ABCPMC si nous appliquions la méthodologie à l'aveugle. Ainsi, il faudrait que l'algorithme ABCPMC analyse tous les facteurs plus influents que celui détérioré et réussisse à retrouver la détérioration malgré la laine minérale qui semble jouer un rôle très majoritaire.

6.2.2 Défauts non repérables

Nous avons ensuite les défauts qui n'ont pas été repérés par l'analyse de Morris comme facteur influent sur la variation de la température intérieure : le béton d'argile, la laine de verre, les hourdis et la dalle béton. Pour chacun de ces défauts, les résultats sont présentés respectivement que les figures III.36, III.37, III.38 et III.39.

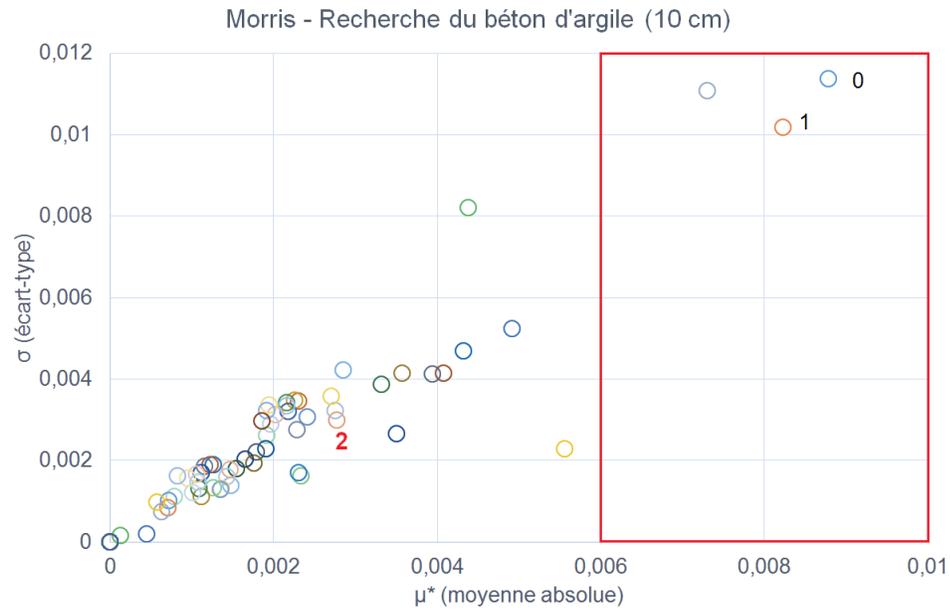


Figure III.36 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

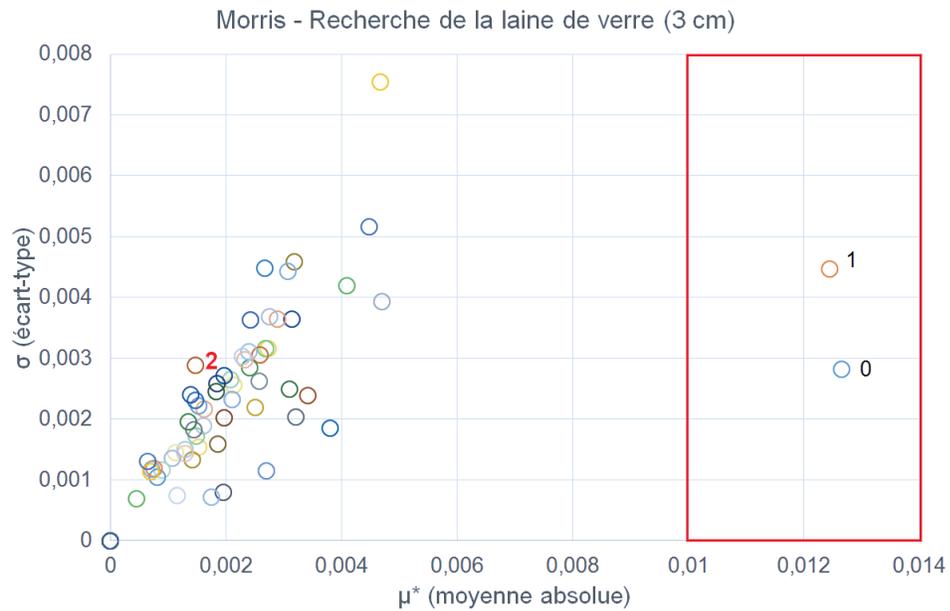


Figure III.37 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

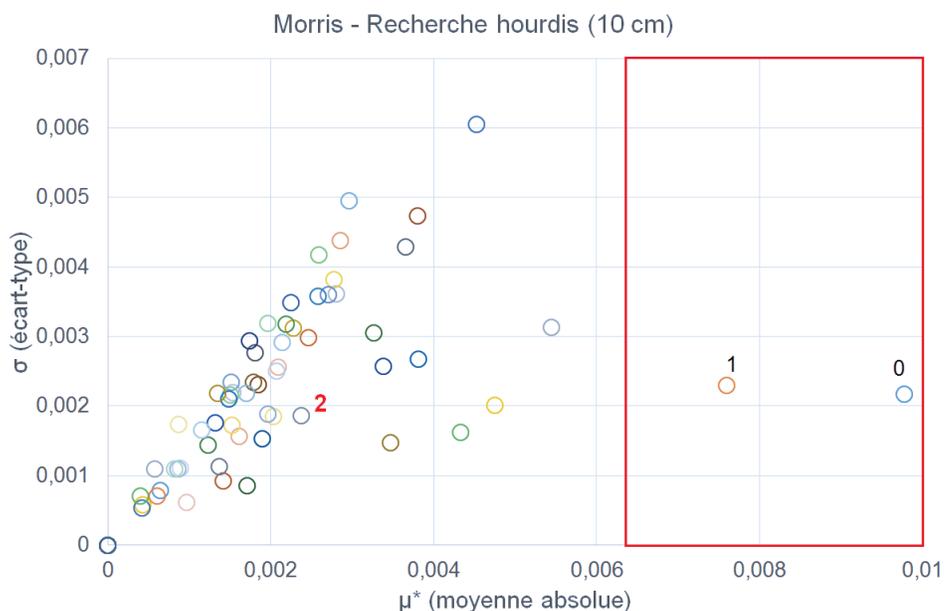


Figure III.38 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

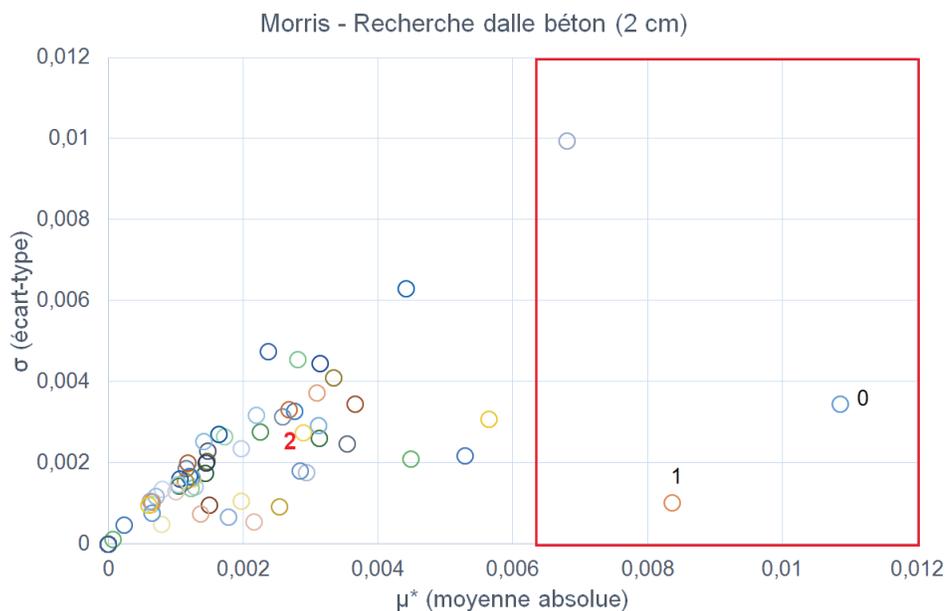


Figure III.39 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

Dans chacun de ces cas, nous pouvons remarquer que la laine minérale (« 0 » et « 1 ») prédomine largement et est le facteur le plus influent qu'importe la source de l'altération engendrée. A chaque fois, notre défaut (« 2 ») fait partie des facteurs négligeables et ainsi, ne serait pas retenu pour l'application de l'algorithme ABCPMC si nous effectuions cette méthodologie à l'aveugle, c'est-à-dire sans savoir quel défaut a été introduit dans le deuxième modèle. Il semble donc que ces défauts ne soient pas discriminables et que la notion de

discriminabilité des défauts qui peuvent être recherchés est très importante pour le bon fonctionnement de la méthodologie.

6.3 Lien entre écart de températures et réparabilité avec l'analyse de Morris

Nous avons vu qu'il semblait difficile d'identifier facilement par la méthodologie d'autres défauts qu'une détérioration de la laine minérale. Pour mieux comprendre pourquoi notre méthodologie a plus de difficultés sur d'autres, nous avons alors étudié la différence de températures induite par chacun de ces défauts pour voir de manière simple comment est influencée la température par ces défauts. Cette différence est représentée par la figure III.40. On peut déjà remarquer que les défauts impliquent un ordre de grandeur différent sur l'écart de températures.

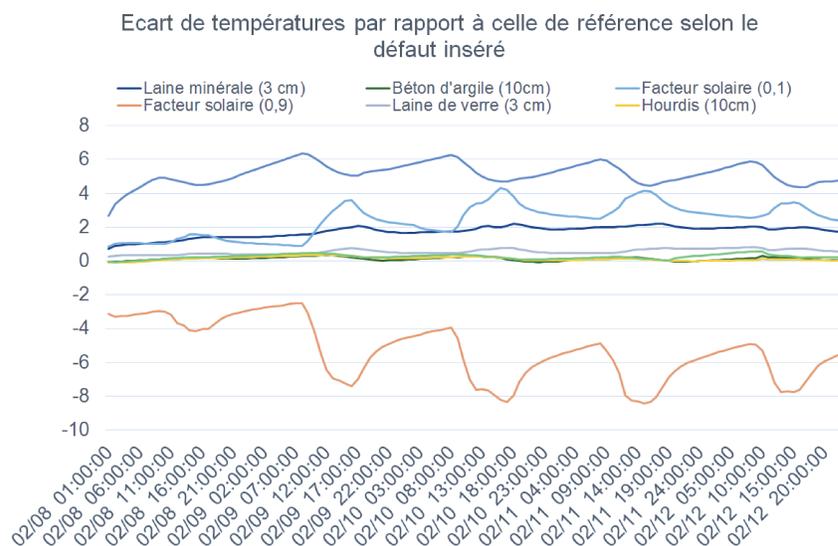


Figure III.40 – Ecart de températures pour l'échantillon hiver selon le défaut inséré.

Si l'on sépare ceux dont l'écart est important et ceux dont il est moindre, nous obtenons respectivement les figures III.41 et III.42. On voit que, sur le premier graphique, l'écart varie jusqu'à 8°C alors qu'il est de l'ordre de 0,9°C pour le deuxième. Nous pouvons également noter que, sur la figure III.41, nous retrouvons les défauts qui ressortaient plus « réparables » lors de l'analyse de Morris et, sur la figure III.42, les défauts qui étaient considérés comme négligeables par l'analyse. Pourtant, on voit que l'écart induit par la laine minérale (figure III.41) est plus faible que celui induit par le facteur solaire à 0,9 par exemple. Il y aurait donc une incohérence entre l'importance de l'écart et l'analyse de Morris.

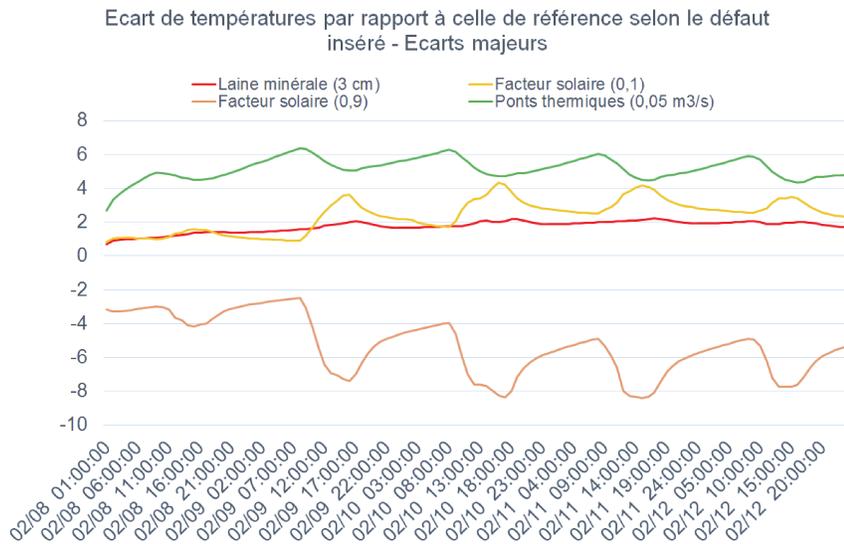


Figure III.41 – Ecart de températures pour l'échantillon hiver selon le défaut inséré - Ecart majeure.

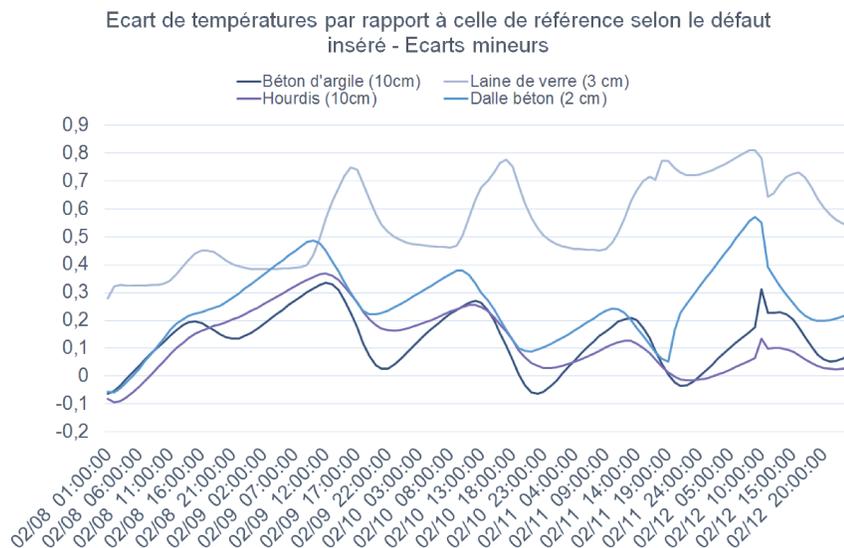


Figure III.42 – Ecart de températures pour l'échantillon hiver selon le défaut inséré - Ecart mineurs.

Si l'on calcule alors le $C_{T_{int}}$ entre la température du modèle initial M0 et celle du modèle modifié M1 pour chacun des défauts, qui est l'indicateur d'écart utilisé lors de l'analyse de Morris, nous obtenons les résultats présentés par la figure III.43. Nous pouvons clairement voir que cet indicateur est très grand lors d'une modification des ponts thermiques, des facteurs solaires et de la laine minérale avec une valeur allant de 60 à 14%. Si les ponts thermiques induisent un écart aussi important, c'est qu'il s'agit de ponts thermiques d'infiltration avec un débit relativement important ($0,05 \text{ m}^3/\text{s}$), introduisant ainsi de l'air froid très rapidement au sein de la maison. La valeur du $C_{T_{int}}$ pour les facteurs solaires s'explique facilement car ils permettent d'augmenter ou de diminuer les apports solaires par les trois fenêtres de la

chambre parentale. Enfin, la laine minérale a également une valeur assez élevée car il s'agit de l'isolant des parois extérieures de la chambre parentale.

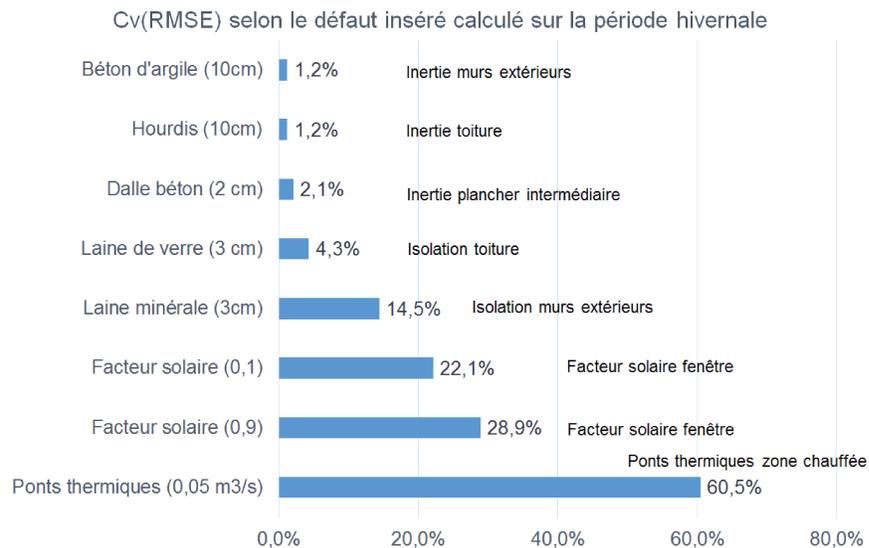


Figure III.43 - $C_{T_{int}}$ pour l'échantillon hiver selon le défaut inséré.

Nous pouvons cependant nous interroger sur l'adéquation de la méthode de Morris pour un problème fin. En effet, nous avons défini des bornes de variation pour chaque facteur étudié lors de l'analyse. Celles-ci sont tirées de la littérature et de l'expérience (Robillart [84]). Or, si nous voulons détecter un défaut plus important, il faudrait étendre ses bornes de variation, tout en étant vigilant à la physique du bâtiment, pour ne pas donner des bornes aberrantes. Ces premières conclusions sont toutefois à étudier plus en détails et peuvent faire l'objet d'analyses futures pour faire le lien entre la discriminabilité des défauts, l'indicateur d'écart $C_{T_{int}}$ et la physique du bâtiment.

7 Conclusion



Dans ce chapitre, nous avons essayé de tester notre méthodologie selon plusieurs critères afin d'en évaluer les limites. Dans la première partie 3, nous avons étudié deux indicateurs, l'écart sur les températures intérieures et l'écart sur les consommations électriques dues au chauffage, et deux périodes sur des saisons différentes. Nous avons ainsi pu voir que, sur une période hivernale, les deux indicateurs donnent de bons résultats, tandis que, sur une période estivale, nous avons testé seulement l'écart de températures, et les résultats ont été plus mitigés du fait d'autres facteurs influençant la température intérieure. Nous en avons donc conclu que la période hivernale semblait plus adéquate à l'application de notre méthodologie.

Dans un deuxième temps (4), nous nous sommes interrogés sur l'aspect temporel afin d'identifier la meilleure durée de simulation pour le modèle (4.1) ainsi que de regarder l'évolution du résultat en fonction de la durée de la période d'absence (4.2), la première permettant d'optimiser le temps de calcul et la seconde permettant d'évaluer la sensibilité du modèle à la durée d'absence. Nous avons montré pour le cas étudié qu'une simulation annuelle n'apportait pas de meilleurs résultats qu'une simulation sur une période plus courte. Nous avons alors retenu une période de simulation commençant 30 jours avant la période d'absence identifiée et finissant à la fin de cette dernière. Nous avons aussi mis en évidence un fait important : l'incertitude sur le résultat, bien qu'acceptable sur une courte période d'absence, devient très faible pour une période d'absence longue (20 à 30 jours).

Dans la troisième partie 5, nous nous sommes penchés sur l'aspect plus spatial en regardant dans un premier temps quel défaut minimal nous pouvions identifier (5.1). Nous avons alors vu qu'à partir d'une épaisseur de 10 cm, le résultat devenait moins précis, notamment à cause d'autres facteurs prenant une influence plus importante avec un défaut moindre.

De ce fait, nous avons ensuite voulu voir ce que donnait une durée d'absence plus longue pour une épaisseur comprise entre 10 et 17 cm, soit un défaut de 1 à 8 cm (5.2). Nous avons donc appliqué l'algorithme ABCPMC pour chaque défaut et pour une durée d'absence comprise entre 1 et 30 jours. Nous avons observé que plus la durée était grande, plus le résultat était précis lorsque l'épaisseur est comprise entre 10 et 13 cm, soit un défaut minimal de 5 cm. Au delà, pour un défaut plus petit, le résultat de l'algorithme ABCPMC tend vers l'épaisseur initiale (18 cm) quelque soit la durée d'absence. Cela peut s'expliquer par le fait qu'il s'agit d'une maison performante et donc qu'un écart de 5 cm sur son isolation n'impacte pas énormément la température intérieure. Cela nous a donc permis d'introduire le concept d'un front de détectabilité « Durée d'absence - Epaisseur cible » séparant les défauts détectables de ceux qui ne le sont pas selon la durée d'absence des occupants et de l'importance du défaut inséré.

Nous nous sommes ensuite intéressés à l'échelle de détectabilité du défaut. En effet, jusqu'à présent nous travaillions sur un défaut global, touchant l'ensemble du bâtiment, et en analysant l'écart de températures dans la chambre parentale. Nous voulions savoir si, pour un défaut global, il était détectable en analysant les autres pièces de la maison (5.3). Dans le cas d'un défaut local, nous voulions savoir jusqu'à quelle échelle nous pouvions aller. En effet, la méthodologie semble fonctionner pour un défaut global, mais est-ce également le cas si le défaut est situé seulement sur la pièce (5.4) voire sur une seule paroi (5.5) ?

Pour le défaut global, nous avons effectué une analyse de Morris avec comme indicateur le $C_{T_{int}}$ de la température intérieure de chacune des pièces, et dans chaque cas, l'isolant ressort. Nous en avons conclu qu'un défaut global est repérable depuis n'importe quelle

pièce. Nous avons ensuite introduit le défaut seulement sur la partie que nous souhaitons étudier. A l'échelle de la pièce, nous avons inséré le défaut d'isolation sur toutes les parois de la chambre parentale et avons laissé le reste des murs tel quel. Nous avons eu un résultat plutôt satisfaisant à la fois lors de l'analyse de Morris, où la laine minérale de la pièce ressort comme facteur influent, et lors de l'application de l'algorithme ABCPMC où une épaisseur d'isolant de 3,39 cm sur la chambre a été retrouvée. A l'échelle de la paroi, nous avons étudié les trois parois donnant sur l'extérieur de la chambre parentale : les façades Nord, Est et Sud. Pour chacune de ces parois, nous avons appliqué notre méthodologie. Pour les façades Sud et Est, les résultats n'ont pas été concluants avec de nombreux facteurs, et notamment les matériaux à inertie, perturbant l'analyse de Morris, ne permettant pas ainsi de détecter le défaut inséré comme facteur influent. Ces façades possèdent des fenêtres qui peuvent éblouir en quelque sorte la méthodologie et donc altérer la discriminabilité du défaut inséré. Pour la façade Nord, les résultats sont bons : l'analyse de Morris indique que l'épaisseur de laine minérale de cette paroi est le facteur le plus influent et l'algorithme ABCPMC retrouve une épaisseur de 3,31 cm.

Enfin, nous avons voulu tester d'autres défauts que l'épaisseur de l'isolant (6). Nous avons effectué une analyse de Morris pour chaque défaut, et la laine minérale continue à ressortir comme facteur le plus influent. Cela nous a amené à nous poser la question de l'adéquation de la méthode de Morris lors de problèmes fins. Comme expliqué ci-dessus, elle nécessite la définition de bornes de variation pour chaque facteur étudié. Celles-ci sont tirées de la littérature et/ou de l'expérience. Or, si nous voulons détecter un défaut plus important, il faudrait étendre ses bornes de variation, tout en étant vigilant à la physique du bâtiment, pour ne pas donner des bornes aberrantes.

Chapitre IV

Vers une première mise en situation de l'approche proposée



Table des matières

1	Introduction	116
2	Choix d'un cas d'études	116
2.1	Maison POS-A : cas théorique valide pour une étude réelle?	116
2.2	Maison Ostwald : plus adéquat?	117
2.2.1	Présentation	117
2.2.2	Plusieurs défauts repérés sur site	118
2.2.3	Choix du défaut à repérer par notre approche	120
3	Multitude de données disponibles	121
3.1	Sélection des mesures disponibles et adéquates pour l'application de la méthodologie	121
3.2	Données disponibles... et fiables?	121
4	Application de l'approche proposée et « points relevés »	122
4.1	Analyse des consommations pour l'identification des zones d'absences des occupants	122
4.2	Diminution des incertitudes par la mise à jour des scénarios d'électricité et du fichier météorologique grâce aux mesures	123
4.2.1	De nouveaux scénarios de consommation d'électricité	123
4.2.2	Un nouveau fichier météorologique pour le modèle	124

4.3 Etude de l'écart de température intérieure entre le modèle initial, le modèle mis à jour et la mesure	125
4.4 Analyse de sensibilité sur l'écart de température	128
5 Conclusion de l'application de la méthodologie à un cas réel	132
Conclusion	135
Bibliographie	139
Annexes	146
A Options IPMVP	147

Résumé



Dans ce chapitre, nous mettons pour la première fois notre méthodologie en situation d'un cas réel. Nous abordons ici un grand nombre de questions suscitées par cette application. En effet, le cas d'études doit répondre à un certain nombre de critères que nous présentons dans la première partie de ce chapitre. Dans une deuxième partie, nous montrons qu'une attention particulière doit être portée sur les mesures que l'on décide d'utiliser et qu'un suivi minutieux des capteurs in-situ est essentiel pour l'étude d'un cas réel. Nous appliquons, dans la troisième partie de ce chapitre, notre méthodologie, en commençant par l'identification des absences des occupants grâce à l'analyse des consommations puis nous mettons à jour notre modèle selon les consommations mesurées des équipements électriques et les données météorologiques de l'année en cours. Nous étudions ensuite l'écart de température entre celle générée par le modèle et celle mesurée. Si celui-ci est jugé suffisamment important, nous continuons notre méthodologie par une analyse de Morris et l'application de l'algorithme ABCPMC. Dans une dernière partie, nous tirons les conclusions de l'application de la méthodologie développée sur un cas réel, permettant ainsi d'observer ses points forts et ses points faibles qui seront à améliorer par de futurs travaux.

1 Introduction

Dans cette partie, nous proposons de répondre aux questions que l'on doit se poser lors de la mise en situation de notre approche. En effet, les conditions entre un cas théorique, virtuel, et un cas réel varient considérablement : dans le premier, nous maîtrisons toutes les variables et il n'y a aucune incertitude tandis que, dans le second, les incertitudes (occupants, météo, gestion d'énergie, etc...) influent de manière importante sur les résultats que nous pouvons obtenir.

La première partie de ce chapitre est consacrée au choix de la maison qui nous servira de cas d'études (2). Dans la deuxième partie (3), nous faisons état des données disponibles et de celles que nous allons pouvoir utiliser. Une fois le cas d'études et les données choisies, nous appliquons notre méthodologie (4), en établissant une série d'interrogations pour son bon usage. Enfin, nous faisons un bilan (5) de ce que nous a appris la mise en situation de notre approche, avec d'un côté les questions soulevées et de l'autre les points à perfectionner.

2 Choix d'un cas d'études

2.1 Maison POS-A : cas théorique valide pour une étude réelle ?

Nous avons étudié jusqu'à présent la maison POS-A, située dans le sud de la France. Celle-ci nous a permis de développer notre méthodologie et d'en appréhender les points forts et les points faibles. Elle a également été la première finalisée parmi les maisons du projet COMEPOS et a ainsi fait l'objet de nombreuses études. Nous pouvons citer notamment A. Stefanoiu [98] qui s'est concentrée lors de ses travaux de thèse et a calculé le coefficient de transfert de chaleur à travers les parois extérieures de la maison POS-A. Elle a mis en évidence que celui-ci correspondait au coefficient théorique et a donc conclu que la maison avait été correctement construite.

Nous cherchons ici des défauts au niveau des parois, en termes soit d'infiltrations soit de malfaçons internes. La maison POS-A ne convient donc pas à l'application de notre méthodologie lors de la comparaison de la simulation et du modèle. Nous avons alors cherché un autre cas d'études que nous présentons dans la partie suivante.

2.2 Maison Ostwald : plus adéquat ?

Au sein du projet COMEPOS, plusieurs maisons se sont construites lors de l'année 2016 et ont été soumises à des tests préliminaires à la fin de leur construction telles que la maison Ostwald, située à Ostwald près de Strasbourg représentée en figure IV.1. Cette maison a fait l'objet d'une campagne à la caméra thermique et, suite à cette étude, nous avons choisi d'appliquer la méthodologie à cette maison.

2.2.1 Présentation

La maison Ostwald a été construite afin d'accueillir une famille de quatre personnes : 2 adultes et 2 enfants. Elle est composée de deux niveaux sur un vide-sanitaire, et de combles non-aménagés. Elle a été livrée en octobre 2016. La commune d'Ostwald est située en Alsace à 140 mètres d'altitude, donc dans un climat plus froid que la maison POS-A étudiée précédemment, ce qui devrait favoriser la discriminabilité de défauts liés à l'enveloppe.



Figure IV.1 - Maison Ostwald - Façades sud-ouest et sud-est.

Les murs extérieurs ont une résistance thermique totale de $6 \text{ m}^2 \cdot \text{K}/\text{W}$ avec une isolation par l'intérieur. Une description plus détaillée est proposée en annexe F.1.

Le modèle a été fait sous *Sketchup* (figure IV.2a) puis traduit par *Openstudio* pour l'utiliser avec *EnergyPlus* par un partenaire du projet COMEPOS. Le modèle est très détaillé avec une zone thermique par pièce (figure IV.2b).

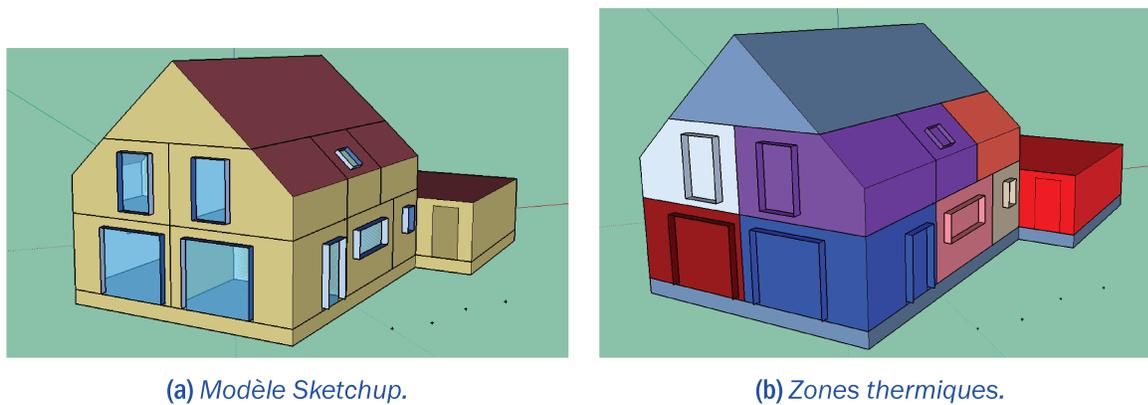


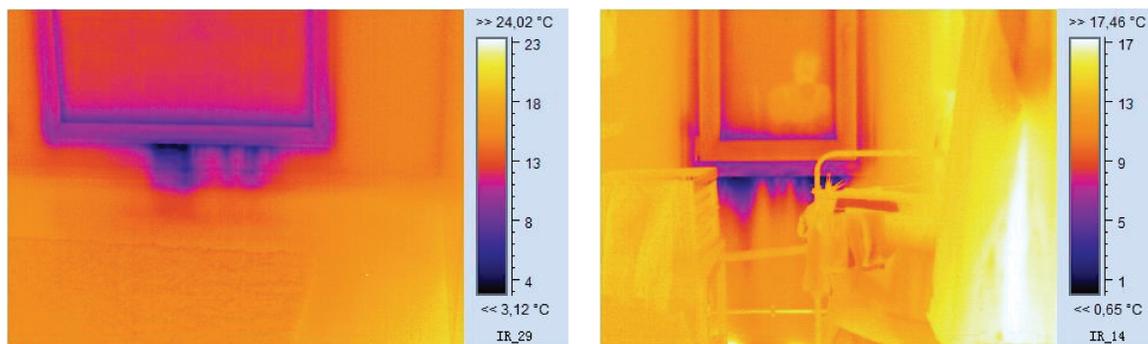
Figure IV.2 – Maison Oswald - Modèle thermique.

2.2.2 Plusieurs défauts repérés sur site

Suite à une campagne à la caméra thermique, plusieurs défauts ont été découverts. Ils peuvent être classés en trois catégories : ouvrants, ponts thermiques et isolation.



(a) Défaut d'étanchéité des caissons des volets roulants.



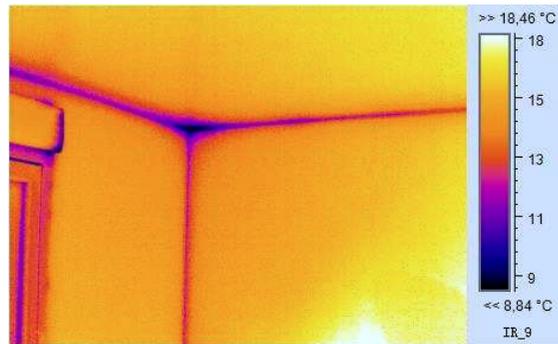
(b) Défaut d'étanchéité des fenêtres, R+1.

(c) Défaut d'étanchéité des fenêtres, RDC.

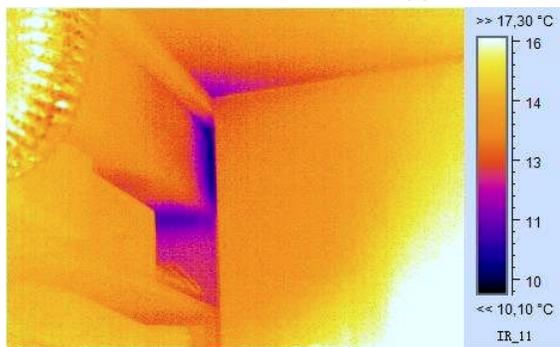
Figure IV.3 – Maison Oswald - Défauts au niveau des ouvrants.

La figure IV.3 présente les défauts de la première catégorie, soit ceux sur les ouvrants. Sur la figure IV.3a, on peut observer un défaut d'étanchéité sur les caissons des volets roulants, qui se retrouve sur tous les caissons de la maison. La figure IV.3b et la figure IV.3c montrent

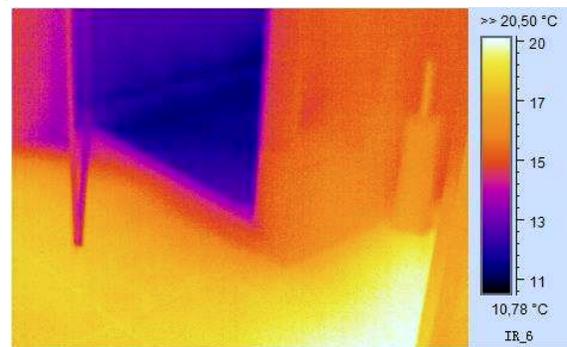
également un défaut d'étanchéité mais cette fois au niveau des fenêtres, respectivement au R+1 et au RDC.



(a) Ponts thermiques dûs à l'ITI.



(b) Ponts thermiques de traversée de poutre.



(c) Déperditions au niveau du bac de douche.

Figure IV.4 – Maison Ostwald - Ponts thermiques.

La figure IV.4 présente les défauts de la deuxième catégorie : les ponts thermiques. On peut observer trois sortes de ponts thermiques majeurs : ceux entre parois (IV.4a), ceux lors de la présence de poutre (IV.4b) et ceux causés par les sanitaires (IV.4c).

La figure IV.5 présente les défauts d'isolation pour chacune des pièces de l'étage. Les figures IV.5a, IV.5b et IV.5c et IV.5d montrent respectivement le défaut dans la chambre parentale, ceux dans les chambres des enfants et celui dans le bureau.

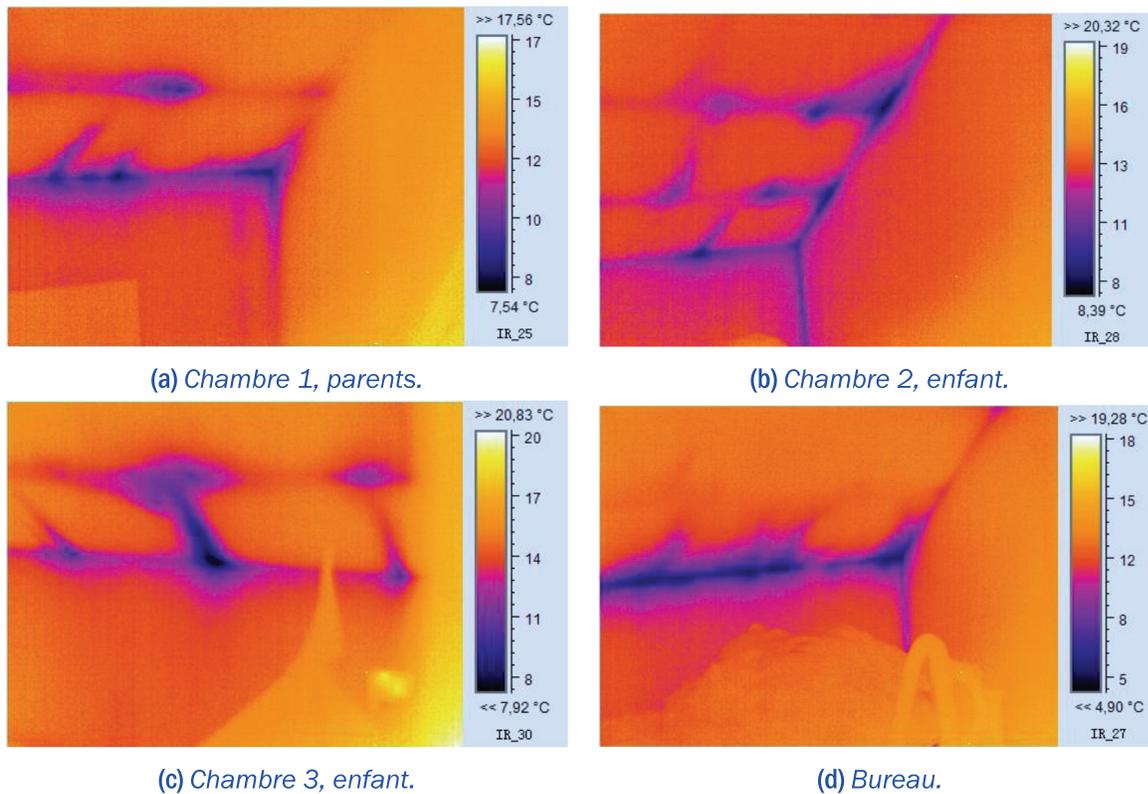


Figure IV.5 – Maison Ostwald - Défaut d'isolation des rampants.

2.2.3 Choix du défaut à repérer par notre approche

Nous avons à présent mis trois catégories de défauts en avant mais toutes ne sont pas idéales pour l'application de notre méthodologie. En effet, travailler sur les ouvrants est une étude fine qui nécessite une connaissance poussée des conditions intérieures et extérieures, ce qui n'est pas toujours le cas lors de l'étude sur une maison « réelle ». En ce qui concerne les ponts thermiques, ceux-ci sont représentés dans *EnergyPlus* par des infiltrations. Nous avons vu dans la dernière partie du chapitre précédent (III.6) que, malgré des infiltrations importantes, ce défaut n'était pas suffisamment influent pour être retenu par l'analyse de Morris. Il semble donc plutôt ambitieux d'initier les essais sur cas réel sur ce type de défauts. Enfin, il y a les déperditions au niveau des rampants. Celles-ci peuvent être dues soit à un défaut d'isolation, soit à des infiltrations. Nous allons dans la suite de ce chapitre nous concentrer sur le défaut d'isolation et essayer de le repérer par l'application de notre méthodologie bien que nous avons vu lors du chapitre III.5.5.2 qu'un défaut très localisé n'était pas évident à localiser déjà dans des conditions connues, maîtrisées.

3 Multitude de données disponibles

Dans le monde du bâtiment connecté, de plus en plus de capteurs sont disposés impliquant de très nombreuses données, mesures, disponibles. La problématique n'est donc plus d'avoir des données mais d'avoir des données dont on peut tirer de l'information. C'est ce que nous allons essayer d'obtenir dans cette partie.

3.1 Sélection des mesures disponibles et adéquates pour l'application de la méthodologie

Dans cette maison, 64 variables sont disponibles grâce à l'installation de plusieurs capteurs¹, permettant ainsi la récupération d'informations sur les températures (intérieures et extérieure), la pression, l'humidité, les consommations électriques et de fluides. La liste de ces variables est disponible en annexe F.2.

Dans le chapitre III, nous avons établi que la méthodologie proposée fonctionnait mieux sur une période d'absence hivernale pour les deux indicateurs étudiés (la consommation et la température intérieure). Il nous a donc fallu nous concentrer sur les données de l'hiver 2016-2017. Cependant, malgré la livraison de la maison en octobre 2016, la plupart des données n'ont été disponibles qu'à partir de mi-janvier 2017, et à partir de fin février pour les consommations électriques. Ces dernières sont essentielles à la méthodologie car elles permettent d'identifier les périodes d'absence.

Les données utilisables sont donc celles à partir de fin février jusque la fin de l'hiver. Dans notre cas, nous souhaitons avoir un mois complet de mesures. Le mois de mars semble donc le mieux convenir à notre application. Selon le climat où se situe la maison considérée, cette période est déjà tardive car la température extérieure va augmenter et devenir plus printanière, ce qui nuit aux résultats de la méthodologie. La maison Ostwald est située en Alsace et, si l'on considère la température extérieure du mois de mars 2017, elle reste relativement froide avec une moyenne à 9,5 °C. Nous nous concentrerons donc sur les données de ce mois pour l'application de notre méthodologie.

3.2 Données disponibles... et fiables ?

Une fois les données sélectionnées, il faut vérifier, avant de les exploiter, leur fiabilité car divers problèmes peuvent apparaître à la suite de la mise en place d'un capteur : il peut

1. Un capteur peut renvoyer plusieurs informations. Il existe par exemple des capteurs pouvant remonter d'un seul coup la température, la pression, le niveau de bruit, l'humidité

renvoyer des données aberrantes qui devront être filtrées, il peut également être défaillant ou déchargé (selon le capteur en question) et il arrêtera alors d'envoyer ses données ou il peut également y avoir un problème au niveau de la remontée de données tel qu'une coupure de la connexion internet.

Ces problèmes amènent alors à des données soit partielles, il faut dans ce cas les reconstruire, soit aberrantes, il faut alors les filtrer. Il est donc nécessaire d'effectuer un prétraitement des données avant de les analyser et de les utiliser dans notre méthodologie.

Après le prétraitement des données, nous nous sommes aperçus que les données de consommation du mois de mars sont inutilisables. En effet, le niveau de consommation relevé par les capteurs est extrêmement faible et n'est donc pas réaliste avec les consommations que peut avoir une famille de quatre personnes. Une action a été mise en place et nous avons pu obtenir ces données seulement à partir du mois de mai, ce qui est trop tardif pour l'application de notre méthodologie. Nous allons donc faire comme si les données étaient correctes et l'appliquer malgré tout, afin de voir les résultats que nous pouvons obtenir. Il faut toutefois noter qu'il est très difficile d'avoir des données fiables dès le premier hiver de fonctionnement.

4 Application de l'approche proposée et « points relevés »

Nous allons ici appliquer notre méthodologie à la maison Ostwald, ce qui nous amènera à réfléchir quant aux conditions optimales de son utilisation, aux points de vigilance et aux parties nécessitant un perfectionnement.

4.1 Analyse des consommations pour l'identification des zones d'absences des occupants

Une fois les données mesurées pré-traitées, nous pouvons identifier les périodes d'absence des occupants par l'analyse des données de consommations (II.4.1.2). La figure IV.6 montre l'évolution des consommations lors du mois de mars 2017. On peut observer deux plateaux sur les consommations électriques générales (courbe jaune) et celles dues aux usages (courbe rouge) entre le 3 et 6 mars puis entre le 17 et 20 mars, tandis que les courbes des fluides (gaz et eau) continuent de croître. Il semble donc possible de supposer que les occupants soient absents sur ces périodes mais qu'ils aient gardé une consigne de température, expliquant ainsi la consommation de gaz pour le chauffage.

La température extérieure étant plus froide au début du mois, nous gardons la première période identifiée soit du 3 au 6 mars 2017.

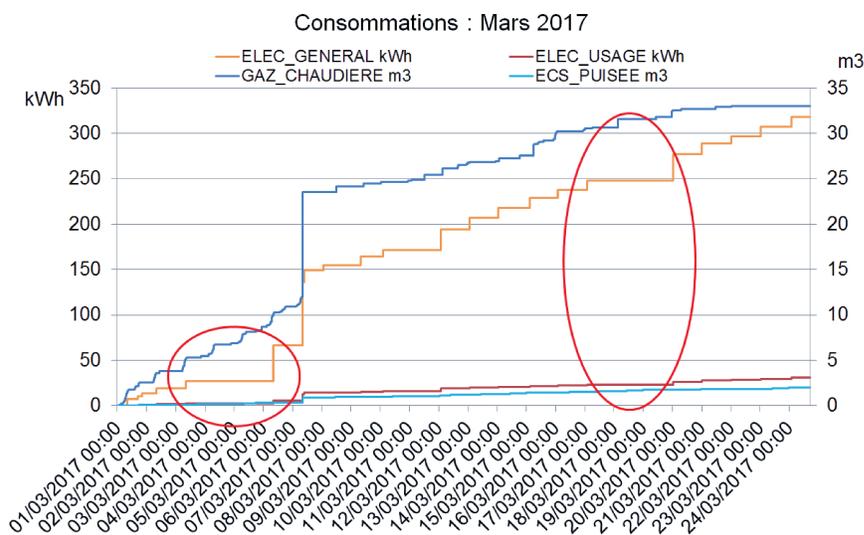


Figure IV.6 – Maison Ostwald - Consommations sur le mois de mars 2017.

4.2 Diminution des incertitudes par la mise à jour des scénarios d'électricité et du fichier météorologique grâce aux mesures

La deuxième étape de la méthodologie consiste à mettre à jour les scénarios de fonctionnement des équipements électriques ainsi que les données météorologiques selon les mesures effectuées sur site. En effet, ces deux facteurs ont une part grandissante dans les performances du bâtiment car, plus celui-ci est performant, et donc bien isolé, plus les apports internes et solaires influent sur la température intérieure et donc sur la consommation énergétique du bâtiment.

4.2.1 De nouveaux scénarios de consommation d'électricité

Dans le cas de la maison Ostwald, il n'y a qu'une variable globale pour la consommation électrique due aux usages, qui correspond ici à la consommation aux différentes prises électriques de la maison. Il y a donc un seul scénario à créer.

L'objet *EnergyPlus* que nous allons utiliser ici est appelé « SCHEDULE :FILE » qui permet de créer un scénario à partir d'un fichier .csv. Il nécessite cependant 8760 entrées soit une par heure. Les données pour l'année 2017 étant partielles au moment de la rédaction de ce manuscrit, nous avons pris celles du mois de mars et les avons dupliquées sur les autres mois de l'année. Nous rappelons ici que les simulations commencent 30 jours avant le début de la période d'absence et se terminent à la fin de l'absence.

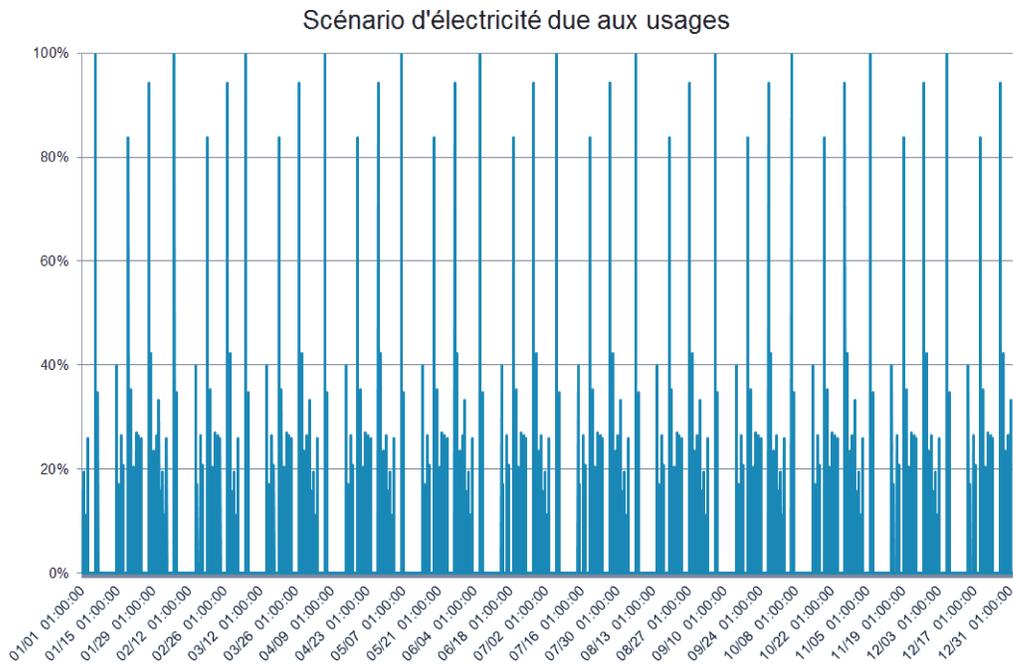


Figure IV.7 – Maison Oswald - Scénario de puissance électrique due à l'usage selon les mesures - $P_{max} = 3.28 \text{ W}$.

La figure IV.7 présente le scénario de consommation des équipements électriques selon les mesures du mois de mars 2017. La consommation maximale mesurée est de 3,28 Wh. Ce niveau de consommation faible nous a alertés sur le bon fonctionnement des capteurs, et après vérification, il s'avère qu'ils étaient défectueux. Nous avons cependant gardé ces mesures afin de continuer d'appliquer notre méthodologie.

4.2.2 Un nouveau fichier météorologique pour le modèle

La deuxième source d'incertitudes non négligeable est les données météorologiques. Celles-ci sont très importantes pour que le modèle soit dans les mêmes conditions que la maison réelle.

Les données météorologiques acquises de la maison Oswald ne sont pas mesurées sur site mais récupérées de la station Strasbourg-Entzheim. Nous disposons de trois fichiers météorologiques, au format .EPW, de cette station. Le premier est issu du site *EnergyPlus*² et il correspond à la météo d'une année type. Le second a été créé à partir des données de *Meteonorm*³. Le troisième et dernier est généré par *Whitebox Technologies*⁴ à partir des mesures de la station pour l'année 2017.

2. https://energyplus.net/weather-location/europe_wmo_region_6/FRA/FRA_Strasbourg.071900_IWEC

3. www.meteonorm.com/

4. <http://weather.whiteboxtechnologies.com/>

La figure IV.8 présente la température extérieure selon ces trois fichiers météo. En noir se trouve la température du fichier météo standard utilisé typiquement lors d'une simulation *EnergyPlus*, en bleu clair, celle générée par le fichier de *Meteonorm*, et enfin, en tirets bleus, celle du fichier *Whitebox*, correspondant aux mesures de la période considérée.

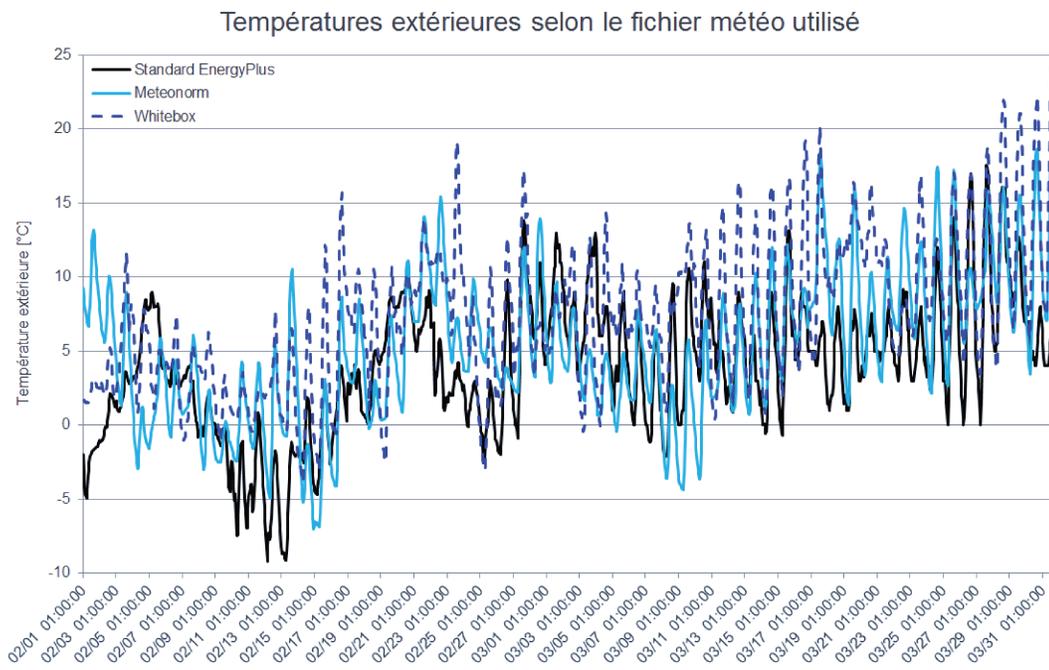


Figure IV.8 – Maison Oswald - Température extérieure selon le fichier météorologique.

On peut remarquer qu'il y a des écarts allant jusque 10°C (25/02) entre deux températures alors qu'elles sont censées représenter la même variable. Il est donc essentiel de prendre le fichier météo qui se rapproche le plus des conditions effectives sur site pour que le modèle puisse représenter au mieux le comportement mesuré de la maison construite. Nous conservons ainsi le fichier généré par *WhiteBox Technologies* pour toutes les simulations effectuées dans la suite de ce manuscrit.

4.3 Etude de l'écart de température intérieure entre le modèle initial, le modèle mis à jour et la mesure

Une fois le modèle mis à jour, il est essentiel d'étudier l'écart de température intérieure avant de procéder à l'analyse de sensibilité (Morris) et au calibrage (ABCPMC).

La figure IV.9 présente la température de la chambre parentale sur la période de simulation selon la modification du modèle effectuée :

— en noir : température initiale,

- en violet : celle lorsque le scénario d'usage est inséré,
- en bleu clair : celle lorsque le nouveau fichier météo est utilisé,
- en rouge clair : température du modèle final avec modification du scénario d'usage et du fichier météo,
- en tirets bleus : température mesurée.

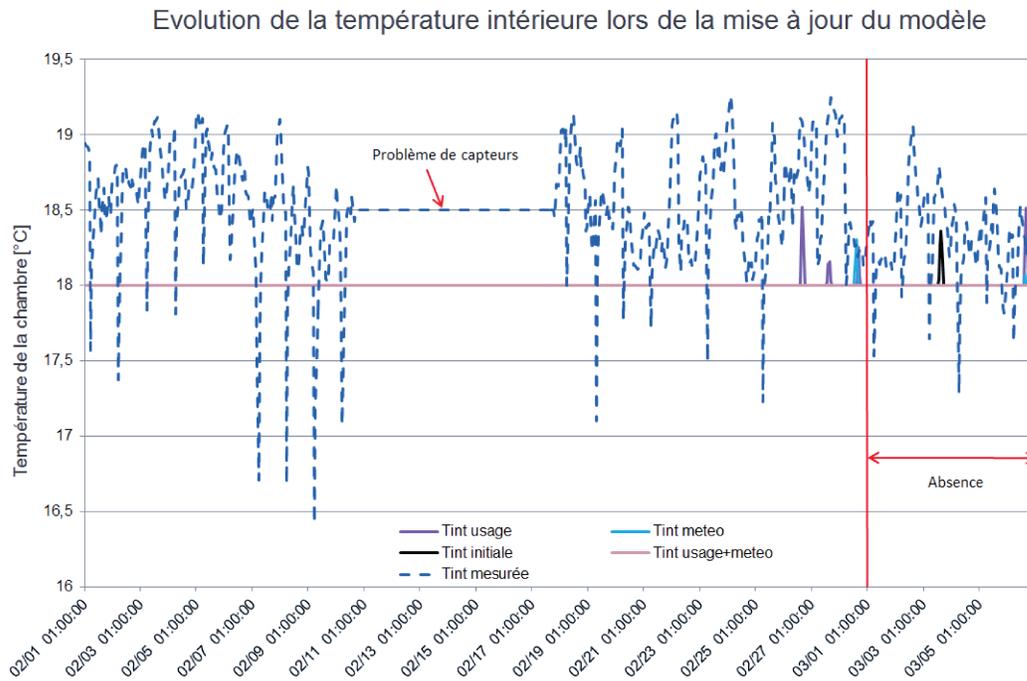


Figure IV.9 – Maison Oswald - Température intérieure en fonction de l'insertion des mesures dans le modèle, comparée à celle mesurée.

Le premier point remarquable c'est que les variations des courbes des températures simulées et celle de la courbe de la température mesurée sont très différentes. Nous n'avons pas accès au système de gestion du chauffage de la maison Oswald, il a donc été impossible de l'insérer dans le modèle. Ce point serait donc à améliorer afin d'avoir des informations plus précises quant au comportement de la maison.

Nous avons donc gardé les consignes initiales de température dans le modèle, soit une température constante à 18°C. Nous pouvons remarquer quelques pics sur les températures simulées en fonction des modifications faites, représentant les différentes excitations auxquelles est soumis le bâtiment lors de ces modifications. En ce qui concerne la température mesurée, on peut observer qu'elle varie globalement entre 16,5 et 19°C, et également un problème au niveau du capteur entre le 11/02 et le 17/11.

Nous avons calculé la MBE (II.1), la RMSE (II.2) et le $C_v(RMSE)$ (II.3) pour la température mesurée et celle après mise à jour du modèle sur la période d'absence considérée soit du 03/03 au 06/03. Les valeurs sont présentées dans le tableau IV.1.

MBE	RMSE	$C_v(RMSE)$
- 1,55%	0,39°C	2,13%

Table IV.1 – Maison Ostwald - Ecart entre la température mesurée et celle simulée après mise à jour du modèle.

Selon la directive 14 de l'ASHRAE, un modèle est calibré si la MBE est inférieure à 10% et le $C_v(RMSE)$ est inférieur à 30% lorsque l'on dispose de données au pas de temps horaire, ce qui est largement notre cas ici.

Ainsi, si l'on n'impose pas de réduct sur la période d'absence, l'écart est a priori trop faible pour que la suite de la méthodologie soit efficace et permette d'avoir une information sur un éventuel défaut, ce qui pose encore une fois la question de la discriminabilité des défauts recherchés. Pour contrer cela, nous allons à présent forcer dans notre modèle un réduct à 7°C sur notre période d'absence. La figure IV.10 présente les différentes températures dans ce cas-ci.

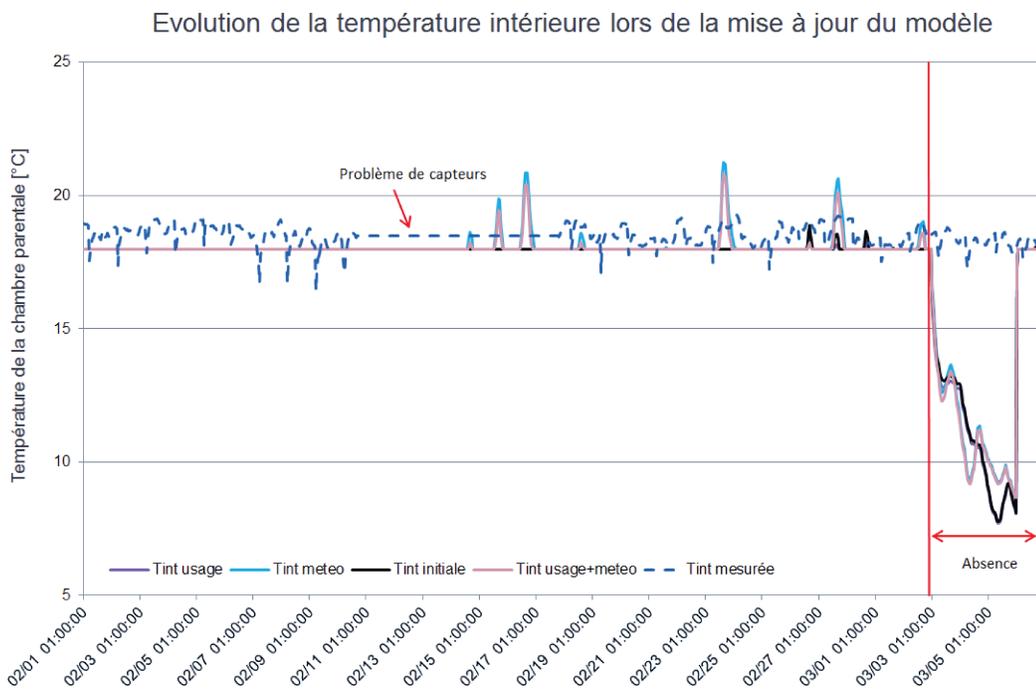


Figure IV.10 – Maison Ostwald - Température intérieure en fonction de l'insertion des mesures dans le modèle, comparée à celle mesurée - Réduit sur la période d'absence.

Nous avons de même calculé les différents indicateurs d'erreur présentés par le tableau IV.2. Nous nous trouvons cette fois dans un cas où le modèle n'est pas calibré sur notre période d'absence ce qui nous permet de continuer d'appliquer la suite de notre méthodologie.

MBE	RMSE	$C_v(RMSE)$
- 30,6%	6,5°C	35,8%

Table IV.2 – *Maison Ostwald - Ecart entre la température mesurée et celle simulée après mise à jour du modèle - Réduit à 7 °C.*

4.4 Analyse de sensibilité sur l'écart de température

Nous effectuons ici une analyse de Morris sur notre modèle mis à jour avec un réduit à 7°C lors de la période d'absence du 03/03 au 06/03. La période de simulation commence le 01/02 et fini le 06/03. Notre indicateur est ici le $C_v(RMSE)$ calculé entre la température mesurée et celle simulée dans la chambre parentale. 133 facteurs sont testés nécessitant 536 simulations. Nous avons identifié les matériaux utilisés dans la chambre parentale et nous nous attendons à les voir ressortir comme facteurs influents par l'analyse de Morris.

La figure IV.11 présente les résultats de cette étude et le tableau IV.3 la légende. On observe que les effets dus aux facteurs 0 et 6 ont à la fois une moyenne absolue et un écart-type élevés, ce qui signifie qu'ils sont non-négligeables et agissent en interaction avec d'autres facteurs ou sont non-linéaires. Les effets des facteurs 1 à 5 ont un écart-type important indiquant également des interactions ou une non-linéarité. On note toutefois que les facteurs 0 et 6 « écrasent » tous les autres facteurs.

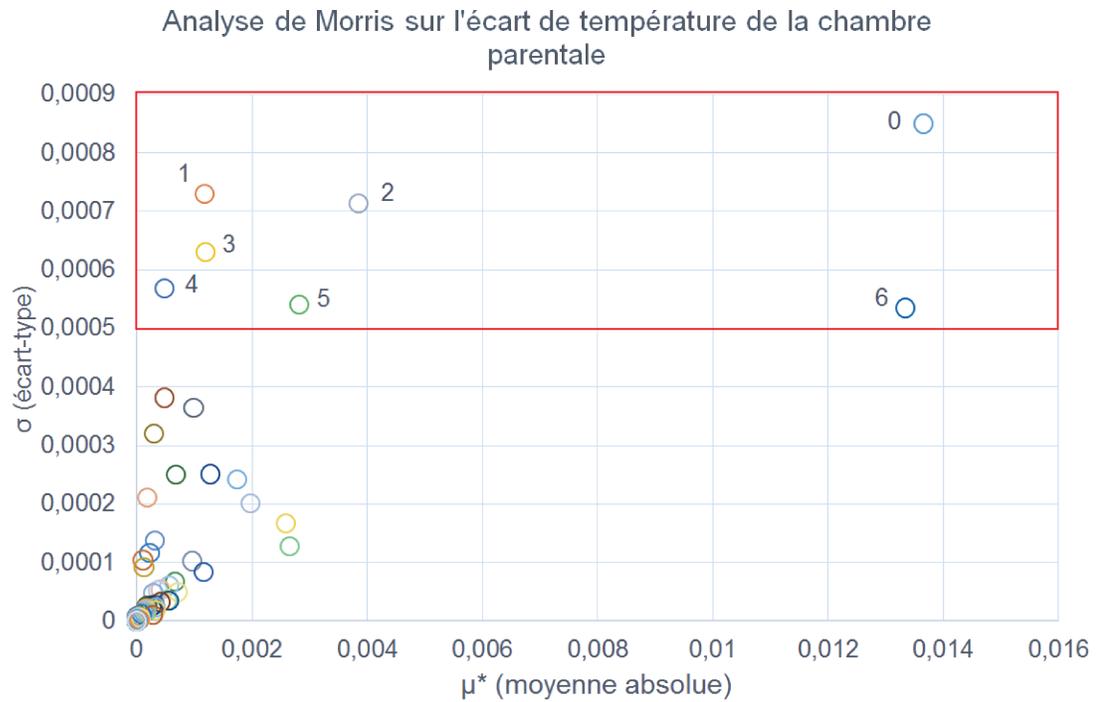


Figure IV.11 – Maison Ostwald - Résultat de l'analyse de Morris sur tous les paramètres.

Index	Facteur
0	Chape - Epaisseur
1	Béton cellulaire - Densité
2	Béton cellulaire - Epaisseur
3	Polystyrène expansé - Conductivité
4	Polystyrène expansé - Epaisseur
5	Béton cellulaire - Conductivité
6	Chape - Densité

Table IV.3 – Légende figure IV.11.

La figure IV.12 permet de localiser les matériaux présents sur les deux parois où des défauts ont été repérés grâce à la caméra thermique (2.2.2). En comparant le tableau IV.3 ci-dessus et cette figure, nous remarquons des corrélations : le béton cellulaire et le polystyrène expansé sont tous deux des matériaux présents dans la composition des murs extérieurs de la maison.

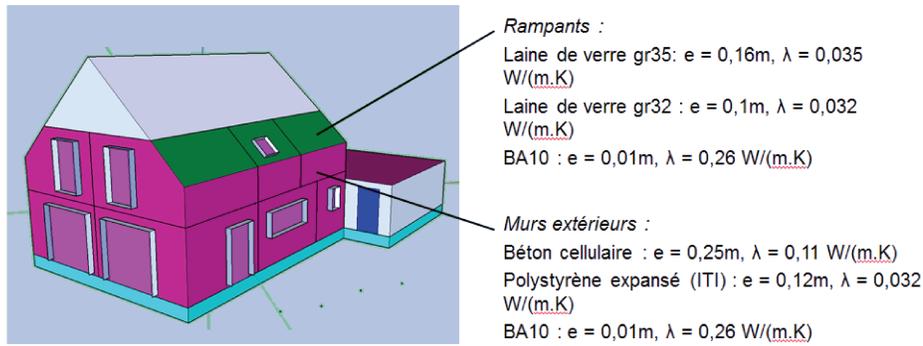


Figure IV.12 – Maison Oswald - Matériaux des rampants et des murs extérieurs.

Cependant, ce ne sont pas ceux de la chambre parentale qui ressortent comme influents mais ceux du reste du bâtiment. Nous allons donc différencier les matériaux pour toutes les pièces de l'étage afin de pouvoir analyser plus finement les effets sur notre écart de température.

La figure IV.13 présente les résultats de cette analyse de Morris. Elle n'est malheureusement pas plus concluante car encore une fois, les matériaux de la chambre parentale ne sont pas reconnus comme facteurs influents. Le tableau IV.4 légende cette figure avec les 10 facteurs les plus influents.

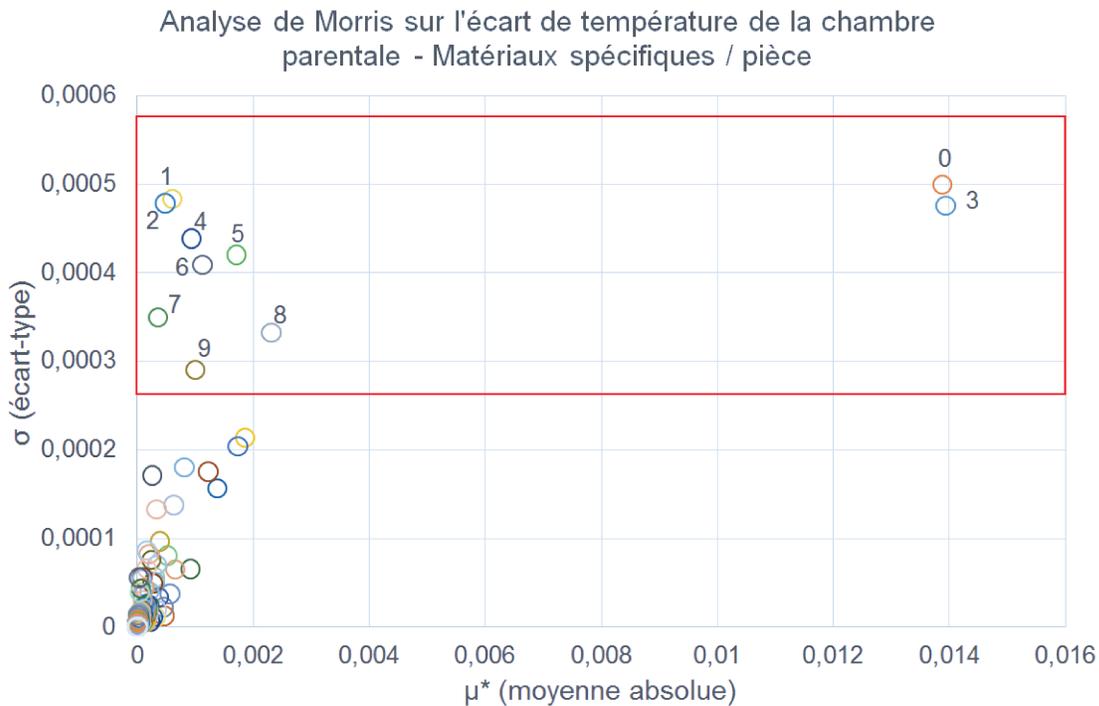


Figure IV.13 – Maison Oswald - Résultat de l'analyse de Morris sur tous les paramètres.

Index	Facteur
0	Chape - Epaisseur
1	Béton cellulaire - Densité
2	Béton cellulaire - Epaisseur
3	Chape - Densité
4	Polystyrène expansé - Densité
5	Béton cellulaire - Conductivité
6	Polystyrène expansé - Epaisseur
7	Polystyrène expansé - Conductivité
8	OSB - Conductivité
9	OSB - Epaisseur

Table IV.4 - Légende figure IV.13.

On observe encore une fois que la chape (facteurs 0 et 3) est le facteur avec l'effet le plus important du fait de son écart-type et de sa moyenne absolue importants. Cela signifie qu'ils sont non-négligeables et agissent en interaction ou sont non-linéaires. Les autres facteurs correspondent à trois matériaux : le béton cellulaire (facteurs 1, 2 et 5) et le polystyrène expansé (facteurs 4, 6 et 7), présents tous deux dans les murs extérieurs, et l'OSB (facteurs 8 et 9), présent dans les combles non aménagés. Nous pouvons noter encore une fois que les matériaux censés être affectés par le défaut repéré à la caméra thermique (2.2.2) ne ressortent pas comme influents. Il semble que la température de la chambre soit plus influencée par la température du RDC que par le défaut local au niveau des rampants.

Nous sommes donc ici face à une impasse car la méthode de Morris ne fait pas ressortir les matériaux situés à l'étage, là où le défaut a été repéré. Cela peut-être dû au fait que le défaut est extrêmement local et n'est donc pas suffisamment important pour qu'il soit détecté. Nous avons en effet vu dans le chapitre précédent d'une part qu'un défaut local n'est pas forcément repérable (III.5.5) et d'autre part qu'il est possible qu'un facteur soit si influent qu'il écrase les effets des autres (III.6). La question de la discriminabilité du défaut semble donc cruciale pour une bonne application de la méthodologie et devra être au coeur des prochains travaux de recherche sur la problématique de détection de défauts à partir de mesures in-situ.

5 Conclusion de l'application de la méthodologie à un cas réel

Dans ce chapitre, nous avons confronté notre méthodologie à un cas réel. Nous avons tout d'abord vu que notre cas d'études théoriques n'était pas utilisable ici. En effet, les performances de la maison POS-A ont déjà été évaluées et aucun défaut sur l'enveloppe n'a pu être identifié. Nous avons donc considéré la maison Ostwald. Des mesures ont été relevées sur l'hiver 2016-2017 et un défaut d'isolation a été repéré à la caméra thermique, ce qui semblait idéal pour l'application de notre méthodologie. La confrontation aux mesures in-situ a permis de mettre en évidence qu'un des points clés pour le bon fonctionnement de notre méthodologie est la fiabilité des données. En effet, il est absolument nécessaire d'effectuer un prétraitement des mesures pour vérifier si les valeurs ne sont pas aberrantes (trop faibles ou trop hautes) et si elles sont complètes car il n'est pas rare qu'il y est soit un problème au niveau du capteur même ou au niveau de la remontée de données comme par exemple une coupure de connexion internet longue. Ce prétraitement nous a permis de mettre en évidence des données de consommation erronées car beaucoup trop faibles par rapport à un usage quotidien de la maison. Nous avons malgré tout continué avec ces données faute d'autre cas d'études pertinent à disposition.

Nous avons alors commencé à appliquer notre méthodologie. La première étape consiste à mettre à jour notre modèle selon la météo et les usages des occupants. Nous avons donc sélectionné un fichier météo fait à partir de mesures du site d'Entzheim, situé à proximité de la maison, et correspondant à la période étudié (hiver 2016-2017) et, à partir des mesures, nous avons créé un nouveau scénario de consommation pour les équipements électriques. Nous avons ensuite identifié une période d'absence entre le 03/03 et le 06/03 grâce aux mesures de consommation.

Nous avons ensuite étudié l'écart de température dans la chambre parentale, située à l'étage. Nous avons observé dans un premier temps un écart très faible entre la température mesurée et celle simulée (IV.1) ce qui nous a amené à forcer un réduct dans notre modèle sur la période d'absence.

Nous avons finalement appliqué la méthode de Morris en identifiant les matériaux situés dans la chambre considérée pour plus de précision. Les résultats ont montré que ces matériaux ne sont pas influents par rapport à la chape du bâtiment mais que des matériaux des murs extérieurs avaient également une certaine influence. Nous avons alors identifié tous les matériaux de l'étage par pièce en espérant trouver l'un d'eux influent. Les résultats de cette analyse ont montré finalement que ce n'était pas les matériaux de l'étage qui avaient le plus

d'influence sur l'écart de température de cette chambre mais les matériaux des pièces du RDC. Cela nous a conforté dans les conclusions que nous avons menées à la fin du chapitre [III](#) quant à la propriété de discrimination de la méthodologie, et notamment sur la capacité de la méthode de Morris à détecter un défaut si localisé, et à la discriminabilité des défauts pour être détecté à distance.

Conclusions et perspectives relatives à l'identification de défauts



Conclusion



Au cours de ces travaux de thèse, nous avons étudié la problématique de la garantie de performance de bâtiments connectés et performants par le biais de l'identification de défauts.

Nous avons pour cela contextualisé notre problème dans l'environnement politique et énergétique actuel. Dans un premier temps, nous avons abordé les enjeux énergétiques et environnementaux liés au secteur du bâtiment et notamment son rôle majeur dans la consommation d'électricité en France. Nous avons ensuite défini le « bâtiment connecté » et les différents usages et développements gravitant autour. Nous nous sommes ensuite intéressés au concept de « garantie de performance » à travers les concepts de la GPEI, garantie de performances énergétiques intrinsèques, et de la GRE, garantie de résultats énergétiques. Nous avons ensuite exposé les moyens mis en oeuvre pour répondre à ces concepts avec notamment le protocole IPMVP définissant des structures de mesure et de vérification des performances, appelées plans M&V. Cela a permis de mettre en évidence des besoins clairs en termes de garantie de performance menant au besoin d'une nouvelle méthode pour l'identification de défauts, sans dérangement pour les occupants et sans déplacement d'une équipe technique.

Nous avons alors répertorié les différentes méthodes pour identifier les défauts dans le bâtiment. celles-ci montrent des résultats satisfaisants mais elles nécessitent toutes un travail sur site (tests) avec des protocoles précis demandant une équipe technique sur place et l'absence des occupants. Nous avons alors développé une nouvelle méthodologie d'identification de défauts tout en palliant à ces problèmes. Elle est composée de deux blocs principaux : le premier vise à réduire les incertitudes dues à l'occupant et à l'environnement extérieur et le second vise à identifier le défaut. Ce dernier est composé de trois étapes clés : (i) un calcul d'écart sur la température intérieure ou bien la consommation électrique qui permet d'évaluer s'il y a un problème en termes de confort ou de coût énergétique, (ii) une analyse de

sensibilité (méthode de Morris) qui permet de réduire le nombre de facteurs potentiellement responsable de l'écart et (iii) un algorithme bayésien (ABCPMC) qui permet de retrouver la loi de probabilité du défaut.

Nous avons ensuite mis à l'épreuve notre méthodologie dans des conditions virtuelles, et donc dans un environnement maîtrisé. Deux modèles strictement identiques ont été considérés et un défaut d'isolant a été introduit dans l'un des deux. Les variables générées par ce modèle représentent ainsi les données mesurées que l'on aurait dans un cas réel.

Nous avons testé différents indicateurs et périodes d'étude pour finalement conclure qu'il est plus adéquat d'appliquer la méthode sur une période hivernale et que, dans ce cas, l'écart sur la température et l'écart sur les consommations électriques sont pertinents comme indicateur. Nous avons alors travaillé seulement sur la période hivernale et avons choisi l'écart de température comme indicateur afin de se placer dans de meilleures conditions de discriminabilité pour notre méthodologie. Nous avons ensuite travaillé sur l'aspect temporel afin d'identifier la meilleure durée de simulation pour le modèle et d'analyser l'évolution du résultat en fonction de la durée de la période d'absence. Nous avons montré que, dans notre cas, une simulation annuelle n'apportait pas de meilleurs résultats qu'une simulation sur une période plus courte.

Nous avons également montré que l'incertitude sur le résultat, bien qu'acceptable sur une courte période d'absence, devient très faible pour une période d'absence longue (20 à 30 jours). Nous avons ensuite étudié l'aspect spatial de la méthode en s'interrogeant sur l'échelle de repérabilité du défaut (à la fois son importance et sa localisation). Sur notre cas d'études (maison POS-A), nous avons montré que le résultat était moins précis à partir d'un défaut d'isolation inférieur à 8 cm. En ce qui concerne sa localisation, nous avons montré que la méthodologie fonctionne pour un défaut global ainsi que s'il n'est présent que sur une pièce. Notre méthode a cependant montré certaines faiblesses lorsque nous avons introduit le défaut seulement sur une paroi, mettant ainsi en évidence que la discriminabilité du défaut doit être prise en compte pour l'application de la méthodologie. Nous avons ensuite voulu créer une typologie de défauts repérables par notre méthodologie mais il s'avère que la méthode de Morris indique à chaque fois l'isolant comme facteur influent, quelque soit le défaut introduit. Cela nous a amené à nous poser la question de l'adéquation de la méthode de Morris lors de problèmes fins.

Nous avons finalement mis notre méthodologie en situation d'un cas réel, ce qui nous a permis de mettre en évidence un certain nombre de questions que l'on doit se poser dans ces conditions. En effet, les mesures doivent être étudiées et analysées avec soin avant d'y appliquer la méthodologie car il n'est pas rare que les mesures soient incomplètes ou qu'elles contiennent des valeurs aberrantes. Après ce pré-traitement, nous avons appliqué notre mé-

thodologie. Nous avons tout d'abord identifié une période d'absence du 03/03 au 06/03, puis nous avons mis à jour notre modèle grâce aux mesures de consommation et à un nouveau fichier météorologique. Une fois cela effectué, nous avons pu commencer à comparer la mesure et la simulation. Nous avons donc étudié l'écart de température entre celle simulée par le modèle mis à jour et celle mesurée dans la chambre parentale à l'étage.

Celui-ci étant extrêmement faible, nous avons forcé un réduct dans notre modèle afin d'augmenter l'écart entre simulation (température réduite) et mesure. Nous avons ensuite appliqué la méthode de Morris afin de voir quels facteurs influent le plus sur l'écart de température. Nous avons pu observer que, selon cette méthode, l'écart est plus sensible aux matériaux des pièces du RDC qu'à ceux présents à l'étage. Ce résultat fait échos à celui obtenu dans le cas virtuel où nous avons observé que la méthode de Morris n'était peut être pas la plus appropriée dans le cas d'un défaut très localisé, peut-être pas suffisamment discriminable.

Ces travaux de thèse ont donc permis le développement d'une première approche, et des outils, pour la détection de défauts à distance basée sur l'analyse de l'écart entre données simulées et données mesurées. Nous en avons apprécié ses points forts et ses points faibles nécessitant donc de futurs travaux que nous allons détailler dans la partie suivante.

Perspectives



Cette méthodologie est une première contribution à l'identification à distance de défauts à partir de mesures in-situ. Il y a donc de nombreuses pistes à explorer pour l'améliorer et pour qu'elle devienne suffisamment robuste.

Le premier point important est l'identification des périodes d'absence. Cette partie devait être simple et facilement implémentable car elle n'est pas le centre des travaux présentés ici mais qui montre quelques limites notamment si les mesures de consommation ne sont pas fiables. Il serait alors utile d'avoir une autre technique permettant d'identifier les absences des occupants.

Nous avons ensuite optimisé pour notre cas virtuel la durée de simulation (III.4.1), à savoir qu'il n'est pas nécessaire d'effectuer une STD sur une année complète mais que commencer la simulation 30 jours avant la période d'absence et la terminer à la fin de cette période suffit. Cet optimum fonctionne pour notre cas et il semble intéressant de le valider sur d'autres cas d'études, voire d'étoffer théoriquement cette hypothèse.

Dans le chapitre III, partie 5.2.2, nous avons proposé le concept de front de discriminabilité de la méthodologie dans l'espace « Durée absence - Epaisseur cible ». Il conviendrait de l'affiner notamment grâce à des études virtuelles complémentaires.

Nous avons vu que, dans le cas réel, il est nécessaire d'avoir accès à la gestion du système de chauffage (planning et consigne de température) afin de pouvoir reproduire le comportement réel dans le modèle. Cela permettrait ainsi d'avoir une température simulée qui correspond mieux à ce qui est censé se passer dans la maison. Le deuxième point d'amélioration serait donc, en se tenant à la contrainte que tout doit être faisable à distance, de développer un programme analysant les différentes variables liées au chauffage et retournant alors son schéma de fonctionnement.

Nous avons observé, à la fois dans le cas virtuel et dans le cas réel, que la méthode de Morris montrait des difficultés lorsque le défaut est très localisé. Il nous semble indispensable de savoir si c'est un problème dû à la méthodologie (qui ne serait pas assez discriminante) ou un problème du défaut (qui n'est pas suffisamment discriminable). Dans le premier cas, des essais d'autres analyses de sensibilité semblent être pertinents pour améliorer la précision de la méthodologie et dans le second cas, nous pourrions envisager des tests « actifs » lors de l'absence des occupants tel que exciter le système volontairement pour rendre les défauts plus discriminables.

Durant ces travaux de thèse, nous avons appliqué la méthodologie à différents cas test et ainsi observé les résultats obtenus pour ces cas précis. Il semble donc nécessaire de généraliser les résultats obtenus, avec appui d'équations mathématiques.

Enfin, nous avons utilisé des modèles fins de simulation thermique dynamique. Ceux-ci sont certes très fins mais ils induisent également un grand nombre d'incertitudes, et un temps de calcul plus important. Ainsi il pourrait être intéressant d'essayer un modèle simplifié afin d'évaluer les améliorations éventuelles.

Bibliographie



- [1] B. ABUSHAKRA, A. SRESHTHAPUTRA, J. S. HABERL, AND D. E. CLARIDGE, *Compilation of Diversity Factors and Schedules for Energy and Cooling Load Calculations*, Tech. Rep. Research Project 1093-RP, ASHRAE, Apr. 2001.
- [2] ADEME, *Climat, Air et Energie : les chiffres clés*, tech. rep., ADEME, 2014.
- [3] ADEME, *Vers un mix électrique 100% renouvelable en 2050*, tech. rep., ADEME, 2015.
- [4] J. AKERET, A. REFREGIER, A. AMARA, S. SEEHARS, AND C. HASNER, *Approximate Bayesian computation for forward modeling in cosmology*, Journal of Cosmology and Astroparticle Physics, (2015).
- [5] F. ALZETTO, D. GOSSARD, AND G. PANDRAUD, *Mesure rapide du coefficient de pertes thermiques des bâtiments*, in Ecobat 2014, Paris, 2014.
- [6] ANR, *Projet INVOLVED ANR-14-CE22-0020*, 2014.
- [7] AQC, *Performance énergétique intrinsèque, PACTE soutient deux projets ambitieux pour développer des outils de mesure à la réception des travaux*, Mar. 2016.
- [8] ARTHUR L., *Bâtiments intelligents : quand l'innovation vient des startups IoT*, Mar. 2016.
- [9] ASHRAE STANDARDS COMMITTEE 2001-2002, *ASHRAE Guideline 14-2002 Measurement of Energy and Demand Savings*, tech. rep., ASHRAE, 2002.
- [10] ASHRAE STANDARDS COMMITTEE 2013-2014, *ASHRAE Guideline 14-2014 Measurement of Energy, Demand, and Water Savings*, tech. rep., ASHRAE, 2014.
- [11] K. ASHTON, *The "Internet of Things" thing*, RFID Journal, (2009).
- [12] P. BACHER AND H. MADSEN, *Identifying suitable models for the heat dynamics of buildings*, Energy and Buildings, 43 (2011), pp. 1511–1522.
- [13] G. BAUWENS AND S. ROELS, *Co-heating test : a state-of-the-art*, in Seminar : Real building energy performance assessment, Belgique, Apr. 2014, pp. 33–46.
- [14] M. A. BEAUMONT, J.-M. CORNUET, J.-M. MARIN, AND C. P. ROBERT, *Adaptive approximate Bayesian computation*, Biometrika, 96 (2009), pp. 983–990. arXiv : 0805.2256.

- [15] M. A. BEAUMONT, C. P. ROBERT, U. P. DAUPHINE, J.-M. MARIN, AND J.-M. CORNUET, *Adaptivity for ABC algorithms : the ABC-PMC scheme*, in In submission, 2008.
- [16] M. A. BEAUMONT, W. ZHANG, AND D. J. BALDING, *Approximate Bayesian Computation in Population Genetics*, *Genetics*, 162 (2002), pp. 2025–2035.
- [17] J. BERGER, S. TASCA-GUERNOUTI, AND M. HUMBERT, *Experimental method to determine the energy envelope performance of buildings*, in 10th International Conference for Enhanced Building Operations, Kuwait, Oct. 2010.
- [18] P. BOISSON AND R. BOUCHIÉ, *ISABELE method : In-Situ Assessment of the Building Envelope performances*, in 9th International Conference on System Simulation in Buildings, Liège, Dec. 2014.
- [19] S. BONTEMPS, *Validation expérimentale de modèles : application aux bâtiments basse consommation*, PhD thesis, Bordeaux, 2015.
- [20] P. BORTOT, S. COLES, AND S. SISSON, *Inference for Stereological Extremes*, *Journal of the American Statistical Association*, 102 (2007), pp. 84–92.
- [21] R. BOUCHIÉ, F. ALZETTO, A. BRUN, C. WEEKS, M. PREECE, M. W. AHMAD, AND M. SISINNI, *Methodologies for the assessment of intrinsic energy performance of buildings' envelop*, rapport technique, Projet PERFORMER, Mar. 2015.
- [22] BOUYGUES IMMOBILIER, *Flexom : Logement connecté*, 2015.
- [23] A. H. BUCKMAN, M. MAYFIELD, AND S. B. M. BECK, *What is a Smart Building ?*, *Smart and Sustainable Built Environment*, (2014).
- [24] F. CAMPOLONGO, J. CARIBONI, AND A. SALTELLI, *An effective screening design for sensitivity analysis of large models*, *Environmental Modelling & Software*, 22 (2007), pp. 1509–1518.
- [25] D. CAVAUD, K. DELAMARE, F.-X. DUSSUD, M. ECOIFFIER, Z. GONG, F. GUGGEMOS, C. HAGÈGE, D. LEPOITTEVIN, J. LAUVERJAT, E. MISAK, J.-P. MARTIN, D. MOMBEL, D. REYNAUD, AND N. RIEDINGER, *Bilan énergétique de la France pour 2015*, tech. rep., Ministère de l'environnement, de l'énergie et de la mer, Nov. 2016.
- [26] C. M. CLEVINGER AND J. HAYMAKER, *The impact of the building occupant on energy modeling simulations*, in Joint International Conference on Computing and Decision Making in Civil and Building Engineering, Montreal, Canada, 2006.
- [27] D. COAKLEY, P. RAFTERY, AND M. KEANE, *A review of methods to match building energy simulation models to measured data*, *Renewable and Sustainable Energy Reviews*, Volume 37 (2014), pp. Pages 123–141.

- [28] D. COAKLEY, P. RAFTERY, P. MOLLOY, AND G. WHITE, *Calibration of a detailed BES model to measured data using an evidence-based analytical optimisation approach*, in Building Simulation, Sydney, Nov. 2011.
- [29] C. COSTA, M. JOUVENT, AND A. DAUGER, *La garantie de performance énergétique*, tech. rep., Plan Bâtiment Durable, Apr. 2012.
- [30] D. B. CRAWLEY, L. K. LAWRIE, F. C. WINKELMANN, W. BUHL, Y. HUANG, C. O. PEDERSEN, R. K. STRAND, R. J. LIESEN, D. E. FISHER, M. J. WITTE, AND J. GLAZER, *EnergyPlus : creating a new-generation building energy simulation program*, Energy and Buildings, 33 (2001), pp. 319–331.
- [31] K. CSILLÉRY, O. FRANÇOIS, AND M. G. B. BLUM, *abc : an R package for approximate Bayesian computation (ABC) : R package : abc*, Methods in Ecology and Evolution, 3 (2012), pp. 475–479.
- [32] A. DARGAHI, *Gestion des flux multi-énergie pour les systèmes V2H*, PhD thesis, Université Grenoble Alpes, Sept. 2014.
- [33] M. DE WIT, *Uncertainty in predictions of thermal comfort in buildings*, PhD thesis, TU Delft, 2001.
- [34] M. DE WIT AND G. AUGENBROEC, *Analysis of uncertainty in building design evaluations and its implications*, Energy and Buildings, Volume 34 (2002), pp. Pages 951–958.
- [35] DELTA DORE, *Delta Dore : Solutions domotiques pour une Maison connectée - Delta Dore*.
- [36] S. DEROUINEAU, J.-M. ALESSANDRINI, P. BOISSON, R. BOUCHILLE, F. BOUGRAIN, J. HANS, R. LAHRECH, S. LAURENCEAU, J.-R. MILLET, AND S. NIBEL, *Bilan de recherche 2011-2013 : Projet Retex, Outils pour le RETour d'EXpérience sur les performances en service des bâtiments de dernière génération*, tech. rep., CSTB, 2013.
- [37] M. DIAGNE, M. DAVID, P. LAURET, J. BOLAND, AND N. SCHMUTZ, *Review of solar irradiance forecasting methods and a proposition for small-scale insular grids*, Renewable and Sustainable Energy Reviews, 27 (2013), pp. 65–76.
- [38] A. B. DOWNEY, *Think Bayes*, 2012.
- [39] ENERGY PLUS, *Auxiliary EnergyPlus Programs - Extra programs for EnergyPlus*, 2015.
- [40] ERGELIS, *Ergelis*, 2011.
- [41] EUROPEAN COMMISSION, *PERFORMER : Portable, Exhaustive, Reliable, Flexible and Optimized appRoach to Monitoring and Evaluation of building eneRgy performance*, Sept. 2013.
- [42] R. EVERETT, *Rapid thermal calibration of houses.*, Open University Energy Research Group report ERG055, Open University, Milton Keynes, UK., 1985.

- [43] EVO, *Concepts et options pour l'évaluation des économies d'énergie et d'eau*, tech. rep., EVO, Sept. 2010.
- [44] —, *Concepts and Options for Determining Energy and Water Savings*, tech. rep., EVO, Jan. 2012.
- [45] FABERNOVEL INNOVATE, *Alfred, l'application de neo-conciergerie de Nexity récompensée au concours Netexplo! / FABERNOVEL INNOVATE*, June 2015.
- [46] M. F. FELS, *PRISM : An Introduction*, *Energy and Buildings*, 9 (1986), pp. 5–18.
- [47] GA SMART BUILDING, *Gapéo : piloter la performance et le confort de vos bâtiments / GA*, 2016.
- [48] J. GOFFART, *Impact de la variabilité des données météorologiques sur une maison basse consommation. Application des analyses de sensibilité pour les entrées temporelles.*, PhD thesis, Université Grenoble Alpes, 2013.
- [49] C. GORSE, D. MILES-SHENTON, D. FARMER, M. FLETCHER, D. GLEW, A. STAFFORD, AND J. PARKER, *The gap between calculated and real performances : Experiences from the laboratory and field and the measures to address the difference*, in *Seminar : Real building energy performance assessment*, Belgique, Apr. 2014, pp. 7–22.
- [50] Y. HADJ SAÏD, R. JOSSE BURET, P. MARÉCHAL, AND F. WURTZ, *Livraison métrologie et début suivi 1ère vague de démonstrateurs*, Tech. Rep. Livrable 5.1.a, COMEPOS, Nov. 2016.
- [51] M. HERRANDO, D. CAMBRA, M. NAVARRO, L. DE LA CRUZ, G. MILLÁN, AND I. ZABALZA, *Energy Performance Certification of Faculty Buildings in Spain : The gap between estimated and real energy consumption*, *Energy Conversion and Management*, 125 (2016), pp. 141–153.
- [52] T. HONG, S. C. TAYLOR-LANGE, S. D'OCA, D. YAN, AND S. P. CORGNATI, *Advances in research and applications of energy-related occupant behavior in buildings*, *Energy and Buildings*, 116 (2016), pp. 694–702.
- [53] M. HUET, M. JOUVENT, AND A. DAUGER, *La garantie de performance énergétique*, tech. rep., Plan Bâtiment Durable, July 2013.
- [54] B. IOOSS, *Analyses d'incertitudes et de sensibilité de modèles complexes - Applications dans des problèmes d'ingénierie*, Mar. 2009.
- [55] —, *Revue sur l'analyse de sensibilité globale de modèles numériques*, *Journal de la Société Française de Statistique*, 152 (2011), pp. 1–23.
- [56] M. G. IPPOLITO, E. R. SANSEVERINO, AND G. ZIZZO, *Impact of building automation control systems and technical building management systems on the energy performance class of residential buildings : An Italian case study*, *Energy and Buildings*, 69 (2014), pp. 33 – 40.

- [57] IZUBA ENERGIES, *Documentation des logiciels IZUBA énergies*.
- [58] F. JABOT, T. FAURE, N. DUMOULIN, AND C. ALBERT, *EasyABC : a R package to perform efficient approximate Bayesian computation sampling schemes*, June 2015.
- [59] C. JIANG, M. K. MASOOD, Y. C. SOH, AND H. LI, *Indoor occupancy estimation from carbon dioxide concentration*, *Energy and Buildings*, 131 (2016), pp. 132–141.
- [60] LE MONITEUR.FR, *Apple fin prêt pour contrôler la maison connectée - Construction Numérique*, Sept. 2016.
- [61] —, *Avec Home, Google veut contrôler la maison connectée - Construction Numérique*, May 2016.
- [62] J. S. LIU AND R. CHEN, *Sequential Monte Carlo Methods for Dynamic Systems*, *Journal of the American Statistical Association*, 93 (1998), pp. 1032–1044.
- [63] K. J. LOMAS AND H. EPPPEL, *Sensitivity analysis techniques for building thermal simulation programs*, *Energy and Buildings*, 19 (1992).
- [64] G. LOWRY, *Modelling user acceptance of building management systems*, *Automation in Construction*, 11 (2002), pp. 695 – 705.
- [65] E. MANGEMATIN, G. PANDRAUD, AND D. ROUX, *Quick measurements of energy efficiency of buildings*, *Comptes Rendus Physique*, 13 (2012), pp. 383–390.
- [66] P. MARJORAM, J. MOLITOR, V. PLAGNOL, AND S. TAVARÉ, *Markov chain Monte Carlo without likelihoods*, *PNAS, Proceedings of the National Academy of Sciences*, 100 (2003), pp. 15324–15328.
- [67] K. MENBERG, Y. HEO, AND R. CHOUDHARY, *Sensitivity analysis methods for building energy models : Comparing computational costs and extractable information*, *Energy and Buildings*, 133 (2016), pp. 433–445.
- [68] J. MEULEMANS, F. ALZETTO, D. FARMER, AND C. GORSE, *QUB/e : A Novel Transient Experimental Method for in situ Measurements of the Thermal Performance of Building Fabrics*, in *Building Information Modelling, Building Performance, Design and Smart Construction*, M. Dastbaz, C. Gorse, and A. Moncaster, eds., Springer International Publishing, 2017, pp. 115–127. DOI : 10.1007/978-3-319-50346-2_9.
- [69] M. D. MORRIS, *Factorial Sampling Plans for Preliminary Computational Experiments*, *Technometrics*, 33 (1991), pp. 161–174.
- [70] F. MUNARETTO, *Etude de l'influence de l'inertie thermique sur les performances énergétiques des bâtiments*, PhD thesis, MINES Paristech, 2014.
- [71] NATIONS UNIES, *Le Protocole de Kyoto*, tech. rep., Nations Unies, 1998.
- [72] NETATMO, *Site Officiel Netatmo® : bienvenue dans votre maison connectée*.

- [73] J. NEYMARK, R. JUDKOFF, G. KNABE, H. T. LE, M. DÜRIG, A. GLASS, AND G. ZWEIFEL, *Applying the building energy simulation test (BESTEST) diagnostic method to verification of space conditioning equipment models used in whole-building energy simulation programs*, *Energy and Buildings*, 34 (2002), pp. 917–931.
- [74] O. ORTEGA, *Les contrats de performance énergétique*, tech. rep., Lefèvre Pelletier & associés, Mar. 2011.
- [75] J. PAGE, *Simulating Occupant Presence and Behaviour in Buildings*, PhD thesis, Ecole Polytechnique Federale de Lausanne, 2007.
- [76] E. PALOMO DEL BARRIO AND G. GUYON, *Using Parameters Space Analysis Techniques for Diagnostic Purposes in the Framework of Empirical Model Validation*, tech. rep., International Energy Agency, Dec. 2002.
- [77] G. PANDRAUD, *Rapid Building Thermal Diagnosis : Presentation of the QUB Method*, Feb. 2014.
- [78] G. PANDRAUD AND R. FITTON, *QUB : Validation of a rapid energy diagnosis method for buildings*, in 4th Expert Meeting, Holzkirchen, Allemagne, Apr. 2013.
- [79] G. PANDRAUD, E. MANGEMATIN, D. ROUX, AND E. QUENTIN, *QUB : a new rapid building energy diagnosis method*, in Proceedings of CLIMA 2013, Prague, République Tchèque, June 2013.
- [80] PARLEMENT EUROPÉEN AND CONSEIL DE L'UNION EUROPÉENNE, *Directive 2012/27/UE*, Oct. 2012.
- [81] J. PRITCHARD, M. SEIELSTAD, A. PEREZ-LEZAUN, AND M. FELDMAN, *Population growth of human Y chromosomes : a study of Y chromosome microsatellites*. - PubMed - NCBI, *Molecular Biology and Evolution*, 16 (1999), pp. 1791–1798. <http://web.stanford.edu/group/pritchardlab/publications/pdfs/PritchardEtAl199b.pdf>.
- [82] C. F. REINHART, *Lightswitch-2002 : a model for manual and automated control of electric lighting and blinds*, *Solar Energy*, 77 (2004), pp. 15–28.
- [83] C. P. ROBERT, *The Bayesian Choice*, Springer Science & Business Media, May 2007.
- [84] M. ROBILLART, *Étude de stratégies de gestion en temps réel pour des bâtiments énergétiquement performants*, PhD thesis, MINES Paristech, 2015.
- [85] S. ROELS, *EBC Annex 58 Reliable Building Energy Performance Characterisation Based on Full Scale Dynamic Measurements*, 2011.
- [86] K. ROSE, S. ELDRIDGE, AND L. CHAPIN, *The Internet of things : an overview*, tech. rep., Internet Society, Oct. 2015.
- [87] M. ROSENBERG AND C. ELEY, *A Stable Whole Building Performance Method for Standard 90.1*, *ASHRAE Journal*, 55 (2013). http://www.eley.com/sites/default/files/pdfs/033-045_Rosenberg_WEB.pdf.

- [88] —, *A Stable Whole Building Performance Method for Standard 90.1-Part II*, ASHRAE Journal, 58 (2016).
- [89] M. ROYAPOOR AND T. ROSKILLY, *Building model calibration using energy and environmental data*, Energy and Buildings, 94 (2015), pp. 109 – 120.
- [90] A. SALTELLI, K. CHAN, AND E. M. SCOTT, *Sensitivity Analysis*, Wiley, Oct. 2008.
- [91] SCHNEIDER ELECTRIC, *Schneider Electric : EcoStruxure*, Apr. 2016.
- [92] O. SIDLER, *L'apport des campagnes de mesures*, in *L'inertie thermique des bâtiments : Consommation et confort d'été en climat méditerranéen*, May 2003.
- [93] —, *Retour d'expériences à partir des missions de suivi et d'évaluation*, Dec. 2009.
- [94] —, *Evaluation par mesure des performances énergétiques des 8 bâtiments construits dans le cadre du programme européen Concerto*, tech. rep., Enertech, Apr. 2011.
- [95] S. A. SISSON, Y. FAN, AND M. M. TANAKA, *Sequential Monte Carlo without likelihoods*, Proceedings of the National Academy of Sciences, 104 (2007), pp. 1760–1765.
- [96] S. SONDEREGGER AND M. MODERA, *Electric Co-heating : A Method for Evaluating Seasonal Heating Efficiencies and Heat Loss Rates in Dwellings*, tech. rep., Energy and Environment Division Lawrence Berkeley Laboratory University of California Berkeley, California, Mar. 1979.
- [97] C. SPITZ, *Analyse de la fiabilité des outils de simulation et des incertitudes de métrologie appliquée à l'efficacité énergétique des bâtiments*, PhD thesis, LOCIE UMR CNRS 5271, Mar. 2012.
- [98] A. M. STEFANOIU, *Vers l'évaluation de la performance des bâtiments à haute efficacité énergétique*, PhD thesis, Université Grenoble Alpes, 2016.
- [99] A. M. STEFANOIU AND P. SCHNEUWLY, *Rapport de simulation et de validation des 10 maisons de la phase 1*, Tech. Rep. Livrable 3.1a, COMEPOS, June 2015.
- [100] S. TAVARÉ, D. J. BALDING, R. C. GRIFFITHS, AND P. DONNELLY, *Inferring Coalescence Times From DNA Sequence Data*, Genetics, 145 (1997), pp. 505–518.
- [101] W. TIAN, *A review of sensitivity analysis methods in building energy analysis*, Renewable and Sustainable Energy Reviews, 20 (2013), pp. 411–419.
- [102] E. VORGER, *Étude de l'influence du comportement des habitants sur la performance énergétique du bâtiment*, phdthesis, Ecole Nationale Supérieure des Mines de Paris, 2014.
- [103] D. WANG, C. C. FEDERSPIEL, AND F. RUBINSTEIN, *Modeling occupancy in single person offices*, Energy and Buildings, 37 (2005), pp. 121–126.
- [104] R. D. WILKINSON, *Approximate Bayesian computation (ABC) gives exact results under the assumption of model error*, Statistical Applications in Genetics and Molecular Biology, 12 (2013), pp. 129–162.

- [105] Y. YAMAGUCHI, Y. SHIMODA, AND M. MIZUNO, *Development of district energy system simulation model based on detailed energy demand model*, in ResearchGate, Netherlands, 2003.
- [106] J. ZHAO, B. LASTERNAS, K. P. LAM, R. YUN, AND V. LOFTNESS, *Occupant behavior and schedule modeling for building energy simulation through office appliance power consumption data mining*, Energy and Buildings, Volume 82 (2014), pp. Pages 341–355.
- [107] P. ZHAO, T. PEFFER, R. NARAYANAMURTHY, G. FIERRO, P. RAFTERY, S. KAAM, AND J. KIM, *Getting into the zone : how the internet of things can improve energy efficiency and demand response in a commercial building*, in Proceedings of ACEEE Summer Study on Energy Efficiency in Buildings., Pacific Grove, CA., Aug. 2016.
- [108] M.-C. ZÉLEM, R. GOURNET, AND C. BESLAY, *Pas de "smart cities" sans "smart habitants"*, Les Cahiers du Développement Urbain Durable, (2013), pp. 45–60.

Annexe A

Options IPMVP



Table des matières

B Présentation de l'outil REFATEC

151

<i>Description de l'option de l'IMPVP</i>	<i>Comment calculer les économies</i>	<i>Applications typiques</i>
OPTION A : Isolement des MCE (Mesure des paramètres clés)		
Les économies sont déterminées par la mesure, sur le terrain, des principaux paramètres de performance énergétique. Les paramètres qui ne sont pas retenus pour être mesurés sont estimés, en se basant par exemple sur des données historiques, des caractéristiques du fabricant, ou encore le jugement de l'expert.	Calcul de la consommation d'énergie en période de référence et en période de suivi, à partir de : (i) Mesures, continues ou sur court terme, des paramètres principaux de fonctionnement, (ii) Valeurs estimées. Ajustements périodiques et non périodiques comme requis.	Dans la mise en place de l'amélioration de l'éclairage, le paramètre principal de la performance est la puissance appelée, laquelle est périodiquement mesurée. Estimation des heures de fonctionnement des appareils d'éclairage, basée sur les horaires d'occupation des bâtiments et le comportement des occupants.

<i>Description de l'option de l'IMPVP</i>	<i>Comment calculer les économies</i>	<i>Applications typiques</i>
OPTION B : Isolement des MCE (Mesure de tous les paramètres)		
Les économies sont déterminées par la mesure, sur le terrain, de la consommation d'énergie des systèmes concernés par les MCE.	Mesure de la consommation d'énergie en période de référence et en période de suivi, ou calculs basés sur des mesures de consommation d'énergie. Ajustements périodiques et non périodiques requis.	Régulation du débit d'une pompe : mesure de la puissance électrique avec un ampèremètre, installé sur l'alimentation électrique du moteur. En période de référence, ce compteur est en place pendant une semaine pour vérifier la constance de la charge. En période de suivi, il mesure les variations de l'utilisation de puissance.
OPTION C : Site entier		
Les économies sont déterminées par la mesure de la consommation d'énergie au niveau du site entier, ou d'une partie autonome de celui-ci. Des mesures, en continu, de la consommation d'énergie du site entier sont effectuées tout au long de la période de suivi.	Analyse des données issues des périodes de référence et de suivi pour le site entier. Ajustements périodiques, si requis, en utilisant des techniques telles que la comparaison simple, ou l'analyse régression. Ajustements non périodiques, si requis.	Programme de gestion de l'énergie, multi-facettes, concernant plusieurs systèmes sur un site. Mesure de consommation d'énergie au moyen de compteurs de gaz et d'électricité des fournisseurs d'énergie, sur une période de référence de 12 mois et pendant toute la période de suivi.

<i>Description de l'option de l'IMPVP</i>	<i>Comment calculer les économies</i>	<i>Applications typiques</i>
OPTION D : Simulation calibrée		
<p>Les économies sont déterminées par la simulation de la consommation d'énergie du site entier, ou d'une partie de celui-ci.</p> <p>Les algorithmes de simulation doivent démontrer une modélisation convenable de la performance énergétique du site.</p> <p>Cette option exige des compétences affirmées dans le domaine de simulation et de calibration.</p>	<p>Simulation de la consommation d'énergie, calibrée au moyen des données horaires ou mensuelles de facturation du fournisseur d'énergie.</p> <p>La mesure de la consommation d'énergie finale peut aider à affiner les données utilisées.</p>	<p>Programme de gestion multi-facettes, concernant plusieurs systèmes sur un site, où il n'existait aucun compteur pendant la période de référence.</p> <p>Les mesures de la consommation d'énergie servent au calibrage de la simulation.</p> <p>La consommation d'énergie en période de référence, déterminée par la simulation calibrée, est comparée à une simulation de la consommation d'énergie en période de suivi.</p>

Annexe B

Présentation de l'outil REFATEC



Table des matières

1 Premier bloc : Traitement des mesures	152
1.1 Fonctionnement	152
1.2 Code	152
2 Second bloc : Comparaison mesure et simulation	160
2.1 Fonctionnement	160
2.2 Code	161
C Maison POS-A	205

1 Premier bloc : Traitement des mesures



1.1 Fonctionnement

Nous rappelons ici le fonctionnement du premier bloc expliqué au chapitre II, partie 5.2.

Le programme nommé « Resampling.py » permet le ré-échantonnage de nos mesures au pas de temps horaire et leur reformatage au format *EnergyPlus*. La figure B.1 présente les entrées nécessaires ainsi que la sortie du programme avec les principaux packages utilisés pour le code présent dans la partie suivante.

Notre mesure de température est présente dans le fichier « hanau_temp_ch1.csv ». Elle est ensuite traitée et enregistrée dans un nouveau csv sous le nom « hanau_temp_ch1_eplus.csv »



Figure B.1 – Schéma récapitulatif du fichier *Resampling.py*.

Avant de pouvoir comparer nos mesures avec les sorties de simulation, nous avons créé de nouveaux scénarios de consommation due à l’usage afin que notre modèle soit exposé au même niveau de charges internes, avant la période d’absence, que notre bâtiment réel. Nous avons alors créé le code « ScheduleMesures.py » permettant de faire cela. La figure B.2 présente son fonctionnement.

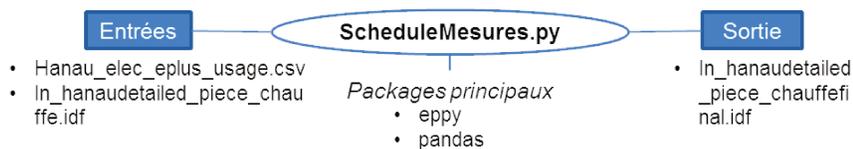


Figure B.2 – Schéma récapitulatif du fichier *ScheduleMesures.py*.

Notre mesure de consommation est présente dans le fichier « hanau_elec_plus_usage.csv ». Notre fichier initial *EnergyPlus* est nommé « in_hanaudetailed_piece_chauffe.idf ». Les mesures sont traitées et insérées dans le fichier .idf sous la forme d’un objet SCHEDULE :FILE. Le fichier .idf modifié est alors enregistré sous le nom « in_hanaudetailed_piece_chauffefinal.idf ».

1.2 Code

Resampling

September 20, 2017

1 IMPORT

```
import pandas as pd
import numpy as np
from datetime import timedelta, datetime
```

2 INITIALISATION

2.1 Noms colonnes

```
sensor_name = 'TEM_CH1_SE_I_001'
equivalent_eplus = u'CH1:Zone Mean Air Temperature [C](Hourly)'
```

2.2 Lecture des mesures

```
df_init = pd.read_csv('Sources/Tint/hanau_temp_ch1.csv', sep= ';')
```

3 Passage de la date au format Datetime

3.1 Modification noms colonnes et transformation en datetime

```
df_init.columns = ['year', 'month', 'day', 'hour', 'minute', sensor_name]
# on selectionne seulement les colonnes avec des indicateurs temps
index_datetime = pd.to_datetime(df_init[df_init.columns[:-1]])
```

3.2 Création d'un nouveau df à 2 colonnes : une avec la date, l'autre avec la température

```
d = {sensor_name : df_init[sensor_name].values}
df_newindex = pd.DataFrame(index = index_datetime, data = d)
```

4 Resampling

4.1 Fonction moyenne pour la température

```
def custom_resampler_mean(array_like):  
    return np.mean(array_like)
```

4.2 Fonctions pour la consommation, les volumes d'eau, etc.

```
## somme si en valeur instantanée  
def custom_resampler_sum(array_like):  
    return np.sum(array_like)  
  
## si compteur : dernière valeur ...  
def custom_resampler_lastvalue(array_like):  
    return array_like.tail(1)  
## ... ou valeur max  
def custom_resampler_max(array_like):  
    return np.max(array_like)
```

4.3 Resampling

```
df_newindex = df_newindex.resample('60T').apply(custom_resampler_mean)  
  
df_newindex.to_csv('Sources/Tint/hanau_tch1_horaire.csv', sep = ';')
```

5 Formatage de la date au format eplus pour comparaison avec simulation

5.1 Fonction pour formatage

```
def eplustimestamp(index_mesures):  
    new_index = []  
    for index in index_mesures:  
        if index.hour == 0 :  
            index_daybefore = index - timedelta(days = 1)  
            index = index_daybefore.strftime(u' %m/%d %H:%M:%S')  
            index = index.replace('00:00:00', '24:00:00')  
            new_index.append(index)  
        else :  
            new_index.append(index.strftime(u' %m/%d %H:%M:%S'))  
    return new_index
```

5.2 Formatage et enregistrement

```
eplus_index = eplustimestamp(df_newindex.index)  
d = {equivalent_eplus : df_newindex[sensor_name].values}  
df_final = pd.DataFrame(index = eplus_index, data = d)
```

In [13]: df_final

```
Out[13]:
```

		CH1:Zone Mean Air Temperature [C] (Hourly)
01/18	01:00:00	18.186667
01/18	02:00:00	18.236667
01/18	03:00:00	18.135000
01/18	04:00:00	18.006667
01/18	05:00:00	17.390000
01/18	06:00:00	16.538333
01/18	07:00:00	16.850000
01/18	08:00:00	16.923333
01/18	09:00:00	17.013333
01/18	10:00:00	17.138333
01/18	11:00:00	17.351667
01/18	12:00:00	17.626667
...		...
07/15	01:00:00	25.083333
07/15	02:00:00	25.093333
07/15	03:00:00	25.045000
07/15	04:00:00	25.000000
07/15	05:00:00	24.988333
07/15	06:00:00	25.101667
07/15	07:00:00	23.808333
07/15	08:00:00	23.548333
07/15	09:00:00	23.260000
07/15	10:00:00	23.138333
07/15	11:00:00	23.611667
07/15	12:00:00	23.971667
07/15	13:00:00	24.091667
07/15	14:00:00	24.133333
07/15	15:00:00	24.210000
07/15	16:00:00	24.278333
07/15	17:00:00	24.145000
07/15	18:00:00	24.095000
07/15	19:00:00	24.088333
07/15	20:00:00	24.108333
07/15	21:00:00	24.108333
07/15	22:00:00	24.206667
07/15	23:00:00	24.403390
07/15	24:00:00	24.305437

[4296 rows x 1 columns]

ScheduleMesures

September 20, 2017

1 IMPORT

importation des packages

```
#!/usr/bin/python2.7
#-*- coding: utf-8 -*-

__author__ = 'rozenn.josse-buret@g2elab.grenoble-inp.fr'

import sys
import os
import shutil
from eppy.modeleditor import IDF
import pandas as pd
import calendar
import datetime
import numpy as np
from os.path import basename, splitext, dirname
from energyplus_wrapper import run
#### LIGNES MAGIQUES POUR LES ACCENTS
import sys
reload(sys)
sys.setdefaultencoding('utf8')
#### FIN LIGNES MAGIQUES
```

2 Définition des chemins

```
sourcespath = 'Sources/'
resultspath = 'Results/'
#fichiers sources
csv_elec = 'Elec/hanau_elec_eplus_usage.csv'
idf_file = 'in_hanaudetailed_piece_chauffage.idf'
idd_file = "/usr/local/EnergyPlus-8-4-0/Energy+.idd"
WeatherFile = 'Environment/whitebox_STRASBOURG_ENTZHEIM_17.epw'
#fichiers de sauvegarde
idf_fileout = 'in_hanaudetailed_piece_chauffetest.idf'
idf_filefinal = 'in_hanaudetailed_piece_chauffefinal.idf'
```

3 Lecture du csv de mesures

J'ai besoin : d'un df de mes mesures de conso en % -> utiliser resampling et de créer un schedule_files à partir de ces mesures

```
df_elec = pd.read_csv(sourcespath + csv_elec, sep = ';')
df_elec = df_elec.set_index(u'Unnamed: 0')
df_elec
```

```
Out[3]:
```

schedule_elec_usage	
Unnamed: 0	
01/01 01:00:00	0
01/01 02:00:00	0
01/01 03:00:00	0
01/01 04:00:00	0
01/01 05:00:00	0
01/01 06:00:00	0
01/01 07:00:00	0
01/01 08:00:00	0,158536585
...	...
12/31 24:00:00	0

[8760 rows x 1 columns]

4 Création de l'objet SCHEDULE:FILE dans le fichier .idf

```
IDF.setidname(idd_file)
idf = IDF(sourcespath+idf_file)
#idf.popidfobject('SCHEDULE:FILE', 0)
for column in df_elec.columns:
    idf.newidfobject("SCHEDULE:FILE", Name = column)
idf.saveas(sourcespath+idf_fileout)

idf1 = IDF(sourcespath+idf_fileout)
run_period = idf1.idfobjects["RUNPERIOD"]
schedules_file = idf1.idfobjects["SCHEDULE:FILE"]
schedules_file
```

```
Out[5]: [
    SCHEDULE:FILE,
        schedule_elec_usage,      !- Name
        ,                          !- Schedule Type Limits Name
        ,                          !- File Name
        ,                          !- Column Number
        ,                          !- Rows to Skip at Top
    8760,                          !- Number of Hours of Data
    Comma,                         !- Column Separator
```

```

        No;                                !- Interpolate to Timestep
    ]

for i,schedule in enumerate(schedules_file):
    schedule.Schedule_Type_Limits_Name = 'Any Number'
    schedule.File_Name = sourcespath + csv_elec
    schedule.Column_Number = i+1
    schedule.Rows_to_Skip_at_Top = 1
    schedule.Number_of_Hours_of_Data = int(len(df_elec[schedule.Name]))

idf1.saveas(sourcespath+idf_filefinal)
schedules_file

```

```

Out[7]: [
    SCHEDULE:FILE,
        schedule_elec_usage,                !- Name
    Any Number,                             !- Schedule Type Limits Name
    Sources/Elec/hanau_elec_eplus_usage.csv, !- File Name
    1,                                       !- Column Number
    1,                                       !- Rows to Skip at Top
    8760,                                    !- Number of Hours of Data
    Comma,                                  !- Column Separator
    No;                                      !- Interpolate to Timestep
]

```

```

old_elecequip=idf1.idfobjects['ELECTRICEQUIPMENT']
for i in range(len(old_elecequip)):
    idf1.popidfobject('ELECTRICEQUIPMENT', 0)

idf1.newidfobject('ELECTRICEQUIPMENT', Name = 'elec_usage')
elecequipments = idf1.idfobjects['ELECTRICEQUIPMENT']

for equipment in elecequipments:
    if 'elec_usage' in equipment.Name :
        equipment.Zone_or_ZoneList_Name = 'DayZone'
        equipment.Schedule_Name = 'schedule_elec_usage'
        equipment.Design_Level = 3.28
idf1.saveas(sourcespath+idf_filefinal)

elecequipments

```

```

Out[8]: [
    ELECTRICEQUIPMENT,
        elec_usage,                        !- Name
    DayZone,                              !- Zone or ZoneList Name
    schedule_elec_usage,                  !- Schedule Name
    EquipmentLevel,                      !- Design Level Calculation Method
    3.28,                                 !- Design Level
    ,                                     !- Watts per Zone Floor Area
]

```

```

,                               !- Watts per Person
0.0,                            !- Fraction Latent
0.0,                            !- Fraction Radiant
0.0,                            !- Fraction Lost
General;                         !- EndUse Subcategory
]

```

5 Vérification

le fichier .idf s'exécute correctement ?

```

run_period[0].Begin_Month = 2
run_period[0].Begin_Day_of_Month = 1
run_period[0].End_Month = 3
run_period[0].End_Day_of_Month = 31
idf1.saveas(sourcespath+idf_filefinal)

```

try :

```

df = run(sourcespath + idf_filefinal,
        sourcespath + WeatherFile,
        working_dir = '.',
        prefix = splitext(basename(idf_filefinal))[0],
        idd_file= "/usr/local/EnergyPlus-8-4-0/Energy+.idd",
        docker_tag = '')

```

```

df = df.set_index(u'Date/Time')

```

except :

```

print '----- ERROR eplus -----'
df = run(sourcespath + idf_filefinal,
        sourcespath + WeatherFile,
        working_dir = '.',
        prefix = splitext(basename(idf_filefinal))[0],
        keep_data = True,
        out_dir = ".",
        idd_file= "/usr/local/EnergyPlus-8-4-0/Energy+.idd",
        docker_tag = '')
print "EPLUS FAILED"

```

```

df.to_csv(resultspath+'eplus_apresmajusage.csv')

```

2 Second bloc : Comparaison mesure et simulation △

2.1 Fonctionnement

Nous rappelons ici le fonctionnement du second bloc expliqué au chapitre II, partie 5.2.

Le second bloc permet l'exécution de deux méthodes : la méthode de Morris et l'algorithme ABCPMC.

Nous souhaitons dans un premier temps effectuer la méthode de Morris. Pour cela, il suffit de lancer le fichier « Morris.py ». La figure B.3, à mettre en parallèle avec le code situé dans la partie suivante, explique son fonctionnement lorsque nous avons étudié dans le chapitre III deux modèles. Les entrées nécessaires sont un fichier .csv de mesures (« in3.csv »), un fichier météo .epw correspondant au lieu étudié (« FR-Carpentras.epw ») et notre modèle de bâtiment au format .idf (« in.idf »). Il va demander à l'utilisateur les facteurs à étudier puis les échantillonner grâce à l'appel des fichiers « Class_IDF.py » et « Class_Samples.py ». Il lancera ensuite l'analyse de Morris et générera un tableau final de valeurs avec les indices de Morris σ (II.7), μ^* II.6 et μ II.5.

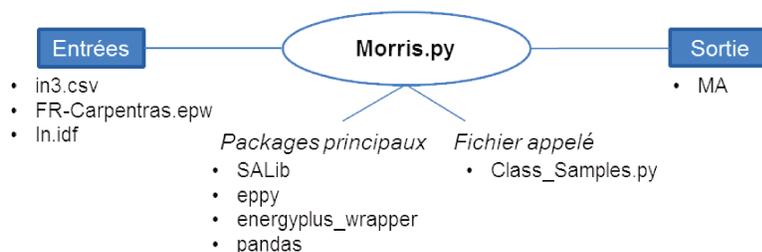


Figure B.3 – Schéma récapitulatif du fichier Morris.py.

Sur le même principe, pour l'exécution de l'algorithme ABCPMC, il suffit de lancer le fichier « Abcpmc.py ». La figure B.4, à mettre en parallèle avec le code situé dans la partie suivante, explique son fonctionnement lorsque nous avons étudié dans le chapitre III deux modèles. Les entrées nécessaires sont toujours un fichier .csv de mesures (« in3.csv »), un fichier météo .epw correspondant au lieu étudié (« FR-Carpentras.epw ») et notre modèle de bâtiment au format .idf (« in.idf »). Il va également demander à l'utilisateur les facteurs à étudier puis les échantillonner toujours grâce à l'appel des fichiers « Class_IDF.py » et « Class_Samples.py ». Il lancera ensuite l'algorithme et générera un objet « pools », où sont stockées toutes les valeurs acceptées pour chaque itération, ainsi qu'un graphique pour chaque paramètre étudié.

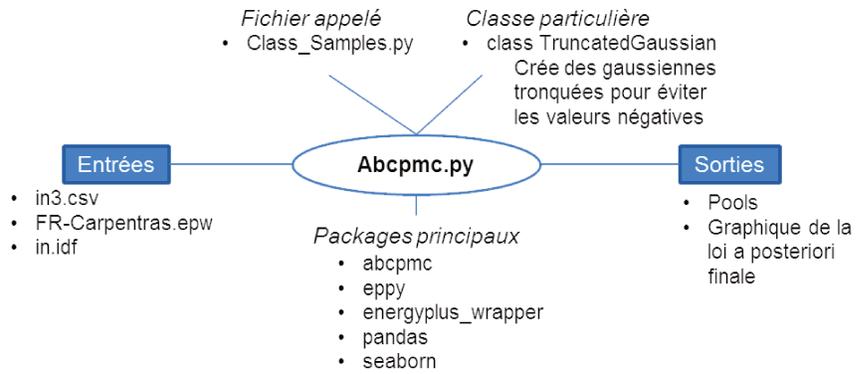


Figure B.4 – Schéma récapitulatif du fichier *Abcpmc.py*.

2.2 Code

Morris

September 13, 2017

1 INITIALISATION

1.1 IMPORT

```
#!/usr/bin/python2.7
#-*- coding: utf-8 -*-

__author__ = 'rozenn.josse-buret@g2elab.grenoble-inp.fr'

import logging
from path import Path
import numpy as np
import matplotlib.pyplot as plt
import subprocess
import traceback
import sys
import pdb
import pandas as pd
import time
from eppy.modeleditor import IDF
import seaborn as sns
#import class_Samples
from scipy import stats, interpolate
from sklearn.metrics import mean_squared_error
from energyplus_wrapper import run
from time import sleep
from os.path import basename, splitext, dirname
import os
from SALib.plotting.morris import horizontal_bar_plot, \
                                     covariance_plot, \
                                     sample_histograms

from class_Samples import Samples
import SALib.analyze.morris as moA
```

1.2 LOGGER

```
logger = logging.getLogger()
logger.handlers = []
```

```

logger.addHandler(logging.FileHandler("/home/rozenn/Documents/EPlusDegrade/\
                                       Wintry/LogFile_EplusFinal.log",mode='w'))
logger.setLevel('INFO')

```

2 FONCTIONS

2.1 MODIFIER LE FICHER EPLUS

```

def modif_eplus(idffile, idffileout, param_set, i):
    idf1 = IDF(idffile)
    materials = idf1.idfobjects["MATERIAL"]
    windows = idf1.idfobjects["WINDOWMATERIAL:SIMPLEGLAZINGSYSTEM"]
    occupants = idf1.idfobjects["PEOPLE"]
    electric equip = idf1.idfobjects["ELECTRICEQUIPMENT"]
    infiltration = idf1.idfobjects["ZONEINFILTRATION:DESIGNFLOWRATE"]
    ventilation = idf1.idfobjects["ZONEVENTILATION:DESIGNFLOWRATE"]
    zones = idf1.idfobjects["ZONE"]

    simul_periodes = idf1.idfobjects["RUNPERIOD"]

    simul_periodes[0].Begin_Month = 1
    simul_periodes[0].Begin_Day_of_Month = 8
    simul_periodes[0].End_Month = 2
    simul_periodes[0].End_Day_of_Month = 13

    dict_error = {}

    ## pour chaque ligne du set de parametres, on modifie le fichier idf
    for key in param_set.columns:
        #print param_set[key][i]
        for material in materials :
            if material.Name.replace(' ', '_') in key :
                if "thickness" in key :
                    if material.Name.replace(' ', '_') == key[:-14]:
                        material.Thickness = param_set[key][i]
                if "conductivity" in key :
                    if material.Name.replace(' ', '_') == key[:-21]:
                        material.Conductivity = param_set[key][i]
                if "specific_Heat" in key :
                    if material.Name.replace(' ', '_') == key[:-23]:
                        material.Specific_Heat = param_set[key][i]
                if "density" in key :
                    if material.Name.replace(' ', '_') == key[:-16]:
                        material.Density = param_set[key][i]

        for window in windows :
            if window.Name.replace(' ', '_') == key :

```

```

        if "Solar_Heat_Gain_Coefficient" in key :
            window.Solar_Heat_Gain_Coefficient = param_set[key][i]
        if "UFactor" in key :
            window.UFactor = param_set[key][i]
        if "Visible_Transmittance" in key :
            window.Visible_Transmittance = param_set[key][i]
    for occupant in occupants :
        if occupant.Name.replace(' ', '_') == key :
            if "People/Area" in key:
                occupant.People_per_Zone_Floor_Area = param_set[key][i]
            if "People" in key and "Area" not in key :
                occupant.Number_of_People = param_set[key][i]
            print "{} {} [{}]: {}".format(occupant.Name, \
                "nb of people",key ,param_set[key][i])

    for electric in electric equip :
        if electric.Name.replace(' ', '_') == key :
            if "Watts/Area" in key :
                electric.Watts_per_Zone_Floor_Area = param_set[key][i]
            if "EquipmentLevel" in key :
                electric.Design_Level = param_set[key][i]
    for infil in infiltration :
        if infil.Name.replace(' ', '_') == key :
            if "flow rate" in key :
                infil.Design_Flow_Rate = param_set[key][i]
            if "air changes" in key :
                infil.Air_Changes_per_Hour = param_set[key][i]
    for ventil in ventilation :
        if ventil.Name.replace(' ', '_') == key :
            if "flow rate" in key :
                ventil.Design_Flow_Rate = param_set[key][i]
            if "air changes" in key :
                ventil.Air_Changes_per_Hour = param_set[key][i]
    # on sauvegarde les modifications de l'idf
    idf1.saveas(idffileout)
return choixOutput(idf1, idffileout)

def choixOutput(idf_file,idfOutput):
    #print "# SUPPRESSION DE L OBJET : OutputControl:IlluminanceMap:Style "
    OutputsIlluminanceMapStyle= idf_file.idfobjects[
        'OUTPUTCONTROL:ILLUMINANCEMAP:STYLE'
    ]
    for i in range(len(OutputsIlluminanceMapStyle)):
        idf_file.popidfobject('OUTPUTCONTROL:ILLUMINANCEMAP:STYLE', 0)

    #print "# SUPPRESSION DE L OBJET : Output:EnvironmentalImpactFactors "
    OutputsEnvironmentalImpactFactors=idf_file.idfobjects[
        'OUTPUT:ENVIRONMENTALIMPACTFACTORS'
    ]

```

```

for i in range(len(OutputsEnvironmentalImpactFactors)):
    idf_file.popidfobject('OUTPUT:ENVIRONMENTALIMPACTFACTORS', 0)

#print "# SUPPRESSION DE L OBJET SOMMAIRE DES TABLES"
OutputsTableSummaryReport=idf_file.idfobjects['OUTPUT:TABLE:SUMMARYREPORTS']
for i in range(len(OutputsTableSummaryReport)):
    idf_file.popidfobject('OUTPUT:TABLE:SUMMARYREPORTS', 0)

#print "# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE Output:Meter"
OutputsMeter=idf_file.idfobjects['OUTPUT:METER']
for i in range(len(OutputsMeter)):
    idf_file.popidfobject('OUTPUT:METER', 0)

#print "# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE Output:Table:TimeBins"
OutputsTableTimeBins=idf_file.idfobjects['OUTPUT:TABLE:TIMEBINS']
for i in range(len(OutputsTableTimeBins)):
    idf_file.popidfobject('OUTPUT:TABLE:TIMEBINS', 0)

#print "# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE OUTPUT:VARIABLE"
OutputsVariable=idf_file.idfobjects['OUTPUT:VARIABLE']
for i in range(len(OutputsVariable)):
    idf_file.popidfobject('OUTPUT:VARIABLE', 0)

## AJOUT DES SORTIES QUE L'ON SOUHAITE OBSERVER ##

idf_file.newidfobject("OUTPUT:TABLE:SUMMARYREPORTS", Report_1_Name = "AllSummary")
# Donnees d'environnement
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Site Outdoor Air Drybulb Temperature",\
    Reporting_Frequency="hourly")

# Donnees batiment

# Temperature
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Zone Mean Air Temperature", \
    Reporting_Frequency="hourly")

# Puissances
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Zone Air System Sensible Heating Rate",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Zone People Sensible Heating Rate",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Zone Air System Sensible Cooling Rate",\
    Reporting_Frequency="hourly")

```

```

idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="Zone Electric Equipment Total Heating Rate",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Water Use Equipment Heating Rate",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="Baseboard Total Heating Rate",\
    Reporting_Frequency="hourly")

#Divers
idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="Baseboard Electric Power",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="People Occupant Count",\
    Reporting_Frequency="hourly")
idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="Zone Lights Electric Power",\
    Reporting_Frequency="hourly")

idf_file.saveas(idfOutput)

return idfOutput

```

2.2 CALCUL DE LA RMSE

```

def dist(x,y):
    """
    x : simulation output / Estimated target values
    y : measurements / Ground truth (correct) target values
    """
    min_periode = ' 02/07 00:00:00' #min_periode.strftime(' %m/%d %H:%M:%S')
    max_periode = ' 02/13 00:00:00' #max_periode.strftime(' %m/%d %H:%M:%S')

    absence_time = [val for val in x.index[1:] if max_periode >= val >= min_periode]
    measured_value_absence = []
    simulated_value_absence = []

    for val in absence_time:
        measured_value = float(y[val])
        measured_value_absence.append(measured_value)
        simulated_value = float(x[val])
        simulated_value_absence.append(simulated_value)

    x = np.array(simulated_value_absence)
    y = np.array(measured_value_absence)
    try:
        mse = mean_squared_error(y, x)

```

```

except ValueError as e:
    logger.error(e)
    logger.error('-- MSE calculation failed')
    raise e
return np.sqrt(mse)/np.mean(y)

```

2.3 COMPTEUR SIMU

```

class counter(object):
    def __init__(self, function):
        self.i = 0
        self.function = function

    def __call__(self, *args, **kwargs):
        self.i += 1
        return self.function(*args, **kwargs)

```

2.4 PLOT MORRIS

```

def plotMorris(MA, MainPath): ### REGLER PROBLEME de SUBPLOT
    # print "##### MA #####\n",MA
    df_dico = pd.DataFrame(MA)
    ## on range pour avoir les parametres les plus influents
    df_dico = df_dico.sort_values(by='mu_star', ascending=False)
    ## en premier dans la legende
    df_dico = df_dico.reset_index()
    df_dico.to_csv(MainPath + "CSV/OUTPUT/MORRIS/morris_Bch_sim30wallmatFinal.csv")

    colors = ['Blue', 'BlueViolet', 'Chocolate', 'Crimson', 'Cyan', 'DarkGreen', \
              'DarkOrange', 'DeepPink', 'DodgerBlue', 'GoldenRod', 'HotPink', \
              'Indigo', 'MidnightBlue', 'OrangeRed', 'Red', 'SlateGrey', \
              'Thistle', 'Black'
              ]
    markers = ['s', 'o', 'D', 'v', '^', '8', '1', '2', '3', '4', \
              'h', 'H', '*', 'x', '+', '+', '>', '<']

    Fig, ax = plt.subplots()
    ax.grid(True) # trace la grille

    fig = plt.gcf()
    fig.canvas.set_window_title('Graph de Morris') # change le nom de la fenetre

    for k in range(0, len(df_dico['names'])):
        ax.plot(df_dico['mu_star'][k],
                df_dico['sigma'][k],
                label=df_dico['names'][k],
                color=colors[k],
                marker=markers[k],

```

```

        markerfacecolor='None',
        markeredgecolor=colors[k],
        linestyle='None') # empeche que des lignes ne relieent les points
ax.annotate(k+1, (df_dico['mu_star'][k], df_dico['sigma'][k]),
            fontsize = 16, xytext = (+5,+5), textcoords = 'offset points')

for label in (ax.get_xticklabels() + ax.get_yticklabels()):
    label.set_fontsize(16)
plt.title("Influence on the Cv(RMSE) between\n simulated \
          and measured operative temperatures", fontsize = 18)

plt.xlabel(ur'$\mu^{\star}$ (absolute mean)', fontsize = 16)
plt.ylabel(ur'$\sigma$ (standard deviation)', fontsize = 16)
plt.savefig(MainPath + "Figures/morris_Bch_sim30wallmatFinal.pdf")
plt.savefig(MainPath + "Figures/morris_Bch_sim30wallmatFinal.png")
plt.show()

```

3 MAIN

3.1 CHEMINS

```

liste = {}
temperature = {}
results = []
run = counter(run)

MainPath = "/home/rozenn/Documents/EPlusDegrade/Wintry/"
OutputPath = MainPath + "CSV/OUTPUT/"
MeasuredData = OutputPath + "in3.csv"
measuredData = pd.read_csv(MeasuredData)
measuredData = measuredData.set_index(u'Date/Time')
choix = u'ETAGE_CH4:Zone Mean Air Temperature [C](Hourly)'
#choix = u'ZONE HVAC BASEBOARD CONVECTIVE ELECTRIC3:\
#Baseboard Total Heating Rate [W](Hourly)'
df_erreur_csv = OutputPath + "df_erreursTintFinal.csv"
EnergyPlusPath = MainPath + 'EnergyplusFiles/'
WeatherFile = EnergyPlusPath + 'WeatherData/FR-Carpentras.epw'
idf_file = EnergyPlusPath + 'in.idf'
idf_fileout = EnergyPlusPath + 'inout.idf'
iddFile = "/usr/local/EnergyPlus-8-4-0/Energy+.idd"
samples = MainPath + "CSV/INPUT/param_set_final_building.csv"

```

3.2 SAMPLES

```

Sample = Samples(MainPath, idf_file, idf_fileout, iddFile, WeatherFile )
problem, N, X, df_samples = Sample.sampling_analyzeMorris()
df_samples.to_csv(samples)

```

4 LANCEMENT DES SIMU

```
print "\n C'EST PARTI POUR "+ str(len(X))+" SIMULATIONS \n\n"
ask = input('\nIl y a {} simulations a effectuer. \
            Souhaitez-vous tout faire ou partitionner \
            (tout = 0 / partie = 1)? : '.format(len(X)))

if ask == 0 :
nb_simul = len(X)
else :
if ask == 1 :
    nb_simul = input('Combien de simulations voulez-vous lancer ? : ')
else :
    nb_simul = 0
    print "Erreur"
depart_simul = 0

results = []
for i in df_samples.index:
    idf_file = modif_eplus(idf_file, idf_fileout, df_samples, i)

try :
    df = run(idf_file,
            WeatherFile,
            working_dir = MainPath,
            prefix = splitext(basename(idf_file))[0],
            idd_file= iddFile,
            docker_tag = '')

    df = df.set_index(u'Date/Time')
except :
    logger.error('----- ERROR eplus -----')
    df = run(idf_file,
            WeatherFile,
            working_dir = MainPath,
            prefix = splitext(basename(idf_file))[0],
            keep_data = True,
            out_dir = "/home/rozenn/Documents/EPlusDegrade/Wintry/eplusFailed/",
            idd_file= iddFile,
            docker_tag = '')
    print "EPLUS FAILED"

## TRAITEMENT DES OUTPUTS
# On lit dans Eppyout.csv la colonne souhaitee
x = df[choix]
y = measuredData[choix]
result = dist(x,y)
results.append(result)
```

4.1 ENREGISTREMENT DES RESULTATS

```
csv_rmse = MainPath + 'csv_testwinter.csv'  
df1 = pd.DataFrame(results)  
df1.to_csv(csv_rmse)
```

4.2 ANALYSE DE MORRIS ET GRAPHIQUES

```
MA = moA.analyze(problem, X, np.array(results), num_resamples=N,  
                 conf_level=0.95, print_to_console=True, grid_jump=2, num_levels=8)  
plotMorris(MA, MainPath)
```

Abcpmc

September 13, 2017

1 INITIALISATION

```
import abcpmc
import logging
from path import Path, tempfile
import numpy as np
import subprocess
import traceback
import sys
import pdb

import pandas as pd
import time
from eppy.modeleditor import IDF
import seaborn as sns
import class_Samples
from scipy import stats, interpolate
from sklearn.metrics import mean_squared_error

from energyplus_wrapper import run
from functools import partial, wraps

#import dill
from pathos.multiprocessing import ProcessPool
import multiprocessing as mp
from multiprocessing import Pool

from contextlib import contextmanager
from os.path import basename, splitext, dirname
import os

    Définitions des chemins et des fichiers utilisés

# Chemins
MainPath = "/home/rozenn/Documents/ABCPMC/"
Path = MainPath + 'abcpmc_Multi/'
ResultsPath = Path + 'Results/'
```

```

SourcesPath = Path + 'Sources/'
# Fichiers
# meteo
WeatherFile = SourcesPath + 'FR-Carpentras.epw'
# eplus file
idffile = 'in.idf'
idffileout = 'inout.idf'
# indicateur etudie
#choix = u'ZONE HVAC BASEBOARD CONVECTIVE ELECTRIC1:\
#Baseboard Total Heating Rate [W](Hourly)'
choix = u'ETAGE_CH4:Zone Mean Air Temperature [C](Hourly)'
# mesures
measures = pd.read_csv(SourcesPath + 'in3.csv', sep = ",")
measures = measures.set_index(u'Date/Time')
y = measures[choix]
# paramètres d'itération du seuil
eps_max = 5
eps_crit = .2
nb_iter_eps = 10
# resultats
ref_results = 'abcpmc_Tint3cm_'
ref_thetas = ResultsPath + ref_results
save_posteriori = ResultsPath + ref_results + 'posteriori.csv'
# fichier log
monlog = MainPath + 'LogFiles/LogFile_EplusMulti.log'

size = 1000

# CLASSES UTILISEES

```

1.1 DECORATEURS

Décorateur Counter() : va compter le nombre d'appels d'une fonction

```

class Counter(object):
    i = mp.Value('i', 1)
    def __init__(self, function):
        self.function = function

    def __call__(self, *args, **kwargs):
        with self.i.get_lock():
            self.i.value += 1
        return self.function(*args, **kwargs)

@property
def counter(self):
    return self.i.value

```

```

def reset(self):
    with self.i.get_lock():
        self.i.value = 0

```

1.2 GAUSSIENNES TRONQUEES

```

class TruncatedGaussianPrior(object):
    """
    Normal gaussian prior

    :param mu: scalar or vector of means
    :param sigma: scalar variance or covariance matrix
    :param mins: scalar or vector of minimum values of the parameters
    :param maxs: scalar or vector of maximum values of the parameters
    """
    def __init__(self, mu, sigma, mins, maxs):
        self.mu = mu
        self.sigma = sigma
        self.mins = mins
        self.maxs = maxs
        self._random = np.random.mtrand.RandomState()

        Nsampling = 5000
        Ninterp = 100

        coordinates = []
        samples = []
        #print "##### DEBUT DU PRIOR : PATIENCE REQUIRED #####"
        for thetai, sigma, Min, Max in zip(mu, sigma.diagonal(), mins, maxs):
            a,b = (Min - thetai) / sigma , (Max - thetai) / sigma
            #print "a et b ", a,b
            samples.append(stats.truncnorm(a,
                                          b,
                                          loc=thetai,
                                          scale=sigma).rvs(size=Nsampling))
            coordinates.append(np.linspace(samples[-1].min(),
                                          samples[-1].max(),
                                          Ninterp)
                               )
        #print "SAMPLES : ", samples
        #print min(samples[0])
        #samples = randomgen(N)

        mesh_coordinates = np.meshgrid(*coordinates)
        positions = np.vstack([X.ravel() for X in mesh_coordinates])
        values = np.vstack(samples)
        kernel = stats.gaussian_kde(values)
        Z = np.reshape(kernel(positions).T, mesh_coordinates[0].shape)

```

```

self.pdf_strict = interpolate.RegularGridInterpolator(coordinates, Z)
self.pdf = kernel
self.randomgen = kernel.resample

#print "randomgen : ", self.randomgen(1).squeeze()

def __call__(self, theta=None):
    if theta is None:
        while True: # Permet une boucle infinie
            randnormal = self.randomgen(1).squeeze()
            #print "randnormal true: ", randnormal
            if all([Min < sample < Max for Min, sample, Max in
                    zip(self.mins, randnormal, self.maxs)]) == True:
                # si la valeur est comprise entre min et max,
                # on renvoie la valeur (et donc on sort de la boucle),
                # dans le cas contraire, on génère une nouvelle valeur
                return randnormal
    else:
        try:
            pdf = self.pdf_strict(theta)[0]
        except ValueError:
            pdf = self.pdf(theta)[0]
        return pdf

```

2 FONCTIONS

2.1 Fermeture de tous les processus

Fonction cPool() :

va nous permettre de fermer tous les processus si interruption du programme. Permet d'éviter les processus "zombie"

<https://gist.github.com/celliern/da7f61e7969f110360c040efc1290fe7>

```

@wraps(Pool)
@contextmanager
def cPool(*args, **kwargs):
    pool = Pool(*args, **kwargs)
    yield pool
    pool.terminate()

```

2.2 Modification du fichier .idf

```

def modif_eplus(idffile, idffileout,ep):
    """
    modification du modele pour creer notre vecteur mesure
    :param idffile : chemin du fichier idf initial a modifier
    :param idffileout : chemin du fichier de sortie
    """

```

```

"""
idf = IDF(idffile)
materials = idf.idfobjects["MATERIAL"]
simul_periodes = idf.idfobjects["RUNPERIOD"]
## pour chaque ligne du set de parametres, on modifie le fichier idf :
## ici seulement l'epaisseur
for material in materials:
    if 'mineral wool' in material.Name:
        material.Thickness = ep

# modification de la periode de simu
simul_periodes[0].Begin_Month = 1
simul_periodes[0].Begin_Day_of_Month = 7
simul_periodes[0].End_Month = 2
simul_periodes[0].End_Day_of_Month = 13

idf.saveas(idffileout)
return choixOutput(idf, idffileout)

def modif_eplus_abc(idffile, param_set):
    """
    modification du modele pour l'algorithme abcpmc
    :param idffile : chemin du fichier idf a modifier
    :param param_set : echantillons pour les differents parametres
    """
    idf = IDF(idffile)
    materials = idf.idfobjects["MATERIAL"]
    windows = idf.idfobjects["WINDOWMATERIAL:SIMPLEGLAZINGSYSTEM"]
    occupants = idf.idfobjects["PEOPLE"]
    electric equip = idf.idfobjects["ELECTRICEQUIPMENT"]
    infiltration = idf.idfobjects["ZONEINFILTRATION:DESIGNFLOWRATE"]
    ventilation = idf.idfobjects["ZONEVENTILATION:DESIGNFLOWRATE"]

    simul_periodes = idf.idfobjects["RUNPERIOD"]
    simul_periodes[0].Begin_Month = 1
    simul_periodes[0].Begin_Day_of_Month = 7
    simul_periodes[0].End_Month = 2
    simul_periodes[0].End_Day_of_Month = 13

    ## pour chaque ligne du set de parametres, on modifie le fichier idf
    for key in param_set.keys():
        #print key
        for material in materials:
            if material.Name.replace(' ', '_') in key:
                if "thickness" in key:
                    material.Thickness = param_set[key]
                    #print material.Thickness
                if "conductivity" in key:

```

```

        material.Conductivity = param_set[key]
    if "specific_Heat" in key:
        material.Specific_Heat = param_set[key]
    if "density" in key:
        material.Density = param_set[key]

for window in windows:
    if window.Name.replace(' ', '_') in key:
        if "Solar_Heat_Gain_Coefficient" in key:
            window.Solar_Heat_Gain_Coefficient = param_set[key]
        if "UFactor" in key:
            window.UFactor = param_set[key]
        if "Visible_Transmittance" in key:
            window.Visible_Transmittance = param_set[key]
for occupant in occupants:
    if occupant.Name.replace(' ', '_') in key:
        if "People/Area" in key:
            occupant.People_per_Zone_Floor_Area = param_set[key]
        if "People" in key and "Area" not in key:
            occupant.Number_of_People = param_set[key]
        print "{} {} [{}]: {}".format(occupant.Name, \
            "nb of people", \
            key, param_set[key][i])

for electric in electric equip:
    if electric.Name.replace(' ', '_') in key:
        if "Watts/Area" in key:
            electric.Watts_per_Zone_Floor_Area = param_set[key]
        if "EquipmentLevel" in key:
            electric.Design_Level = param_set[key]
for infil in infiltration:
    if infil.Name.replace(' ', '_') in key:
        if "flow rate" in key:
            infil.Design_Flow_Rate = param_set[key]
        if "air changes" in key:
            infil.Air_Changes_per_Hour = param_set[key]
for ventil in ventilation:
    if ventil.Name.replace(' ', '_') in key:
        if "flow rate" in key:
            ventil.Design_Flow_Rate = param_set[key]
        if "air changes" in key:
            ventil.Air_Changes_per_Hour = param_set[key]

# on sauvegarde les modifications de l'idf
idf.saveas(idffile)
return choixOutput(idf, idffile)

def choixOutput(idf_file, idfOutput):
    # SUPPRESSION DE L OBJET : OutputControl:IlluminanceMap:Style

```

```

OutputsIllMapStyle= idf_file.idfobjects['OUTPUTCONTROL:ILLUMINANCEMAP:STYLE']
for i in range(len(OutputsIllMapStyle)):
    idf_file.popidfobject('OUTPUTCONTROL:ILLUMINANCEMAP:STYLE', 0)

# SUPPRESSION DE L OBJET : Output:EnvironmentalImpactFactors
OutputsEnvImpactFactors=idf_file.idfobjects['OUTPUT:ENVIRONMENTALIMPACTFACTORS']
for i in range(len(OutputsEnvImpactFactors)):
    idf_file.popidfobject('OUTPUT:ENVIRONMENTALIMPACTFACTORS', 0)

# SUPPRESSION DE L OBJET SOMMAIRE DES TABLES
OutputsTableSummaryReport=idf_file.idfobjects['OUTPUT:TABLE:SUMMARYREPORTS']
for i in range(len(OutputsTableSummaryReport)):
    idf_file.popidfobject('OUTPUT:TABLE:SUMMARYREPORTS', 0)

# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE Output:Meter
OutputsMeter=idf_file.idfobjects['OUTPUT:METER']
for i in range(len(OutputsMeter)):
    idf_file.popidfobject('OUTPUT:METER', 0)

# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE Output:Table:TimeBins
OutputsTableTimeBins=idf_file.idfobjects['OUTPUT:TABLE:TIMEBINS']
for i in range(len(OutputsTableTimeBins)):
    idf_file.popidfobject('OUTPUT:TABLE:TIMEBINS', 0)

# SUPPRESSION DES VARIABLES DE SORTIE DU TYPE OUTPUT:VARIABLE
OutputsVariable=idf_file.idfobjects['OUTPUT:VARIABLE']
for i in range(len(OutputsVariable)):
    idf_file.popidfobject('OUTPUT:VARIABLE', 0)

# AJOUT DES SORTIES QUE L'ON SOUHAITE OBSERVER

idf_file.newidfobject("OUTPUT:TABLE:SUMMARYREPORTS", Report_1_Name = "AllSummary")
# Donnees d'environnement
idf_file.newidfobject("OUTPUT:VARIABLE",\
    Variable_Name="Site Outdoor Air Drybulb Temperature",\
    Reporting_Frequency="hourly")

# Donnees batiment

# Temperature
idf_file.newidfobject("OUTPUT:VARIABLE", \
    Variable_Name="Zone Mean Air Temperature",\
    Reporting_Frequency="hourly")

# Debit dair
#idf_file.newidfobject("OUTPUT:VARIABLE",
    #Variable_Name="Zone Mechanical Ventilation |
        #Air Changes per Hour",\
    #Reporting_Frequency="hourly")

```

```

# Puissances
  #chauffage
idf_file.newidfobject("OUTPUT:VARIABLE", \
                      Variable_Name="Baseboard Total Heating Rate", \
                      Reporting_Frequency="hourly")

#Divers
#idf_file.newidfobject("OUTPUT:VARIABLE", \
                      #Variable_Name="People Occupant Count", \
                      #Reporting_Frequency="hourly")

idf_file.saveas(idfOutput)

return idfOutput

```

2.3 Définition de la distance et du postfn pour ABCPMC

distance : correspond au seuil

postfn : correspond à l'évaluation du modèle

```

def dist(x,y):
    """
    :param x : simulation output / Estimated target values
    :param y : measurements / Ground truth (correct) target values
    """
    min_periode = ' 02/08 01:00:00' #min_periode.strftime('%m/%d %H:%M:%S')
    max_periode = ' 02/12 24:00:00' #max_periode.strftime('%m/%d %H:%M:%S')

    #on selectionne les dates correspondant a la periode d'absence parmi les index
    absence_time = [val for val in x.index[1:] if max_periode >= val >= min_periode]
    measured_value_absence = []
    simulated_value_absence = []

    for val in absence_time:
        measured_value = float(y[val])
        measured_value_absence.append(measured_value)
        simulated_value = float(x[val])
        simulated_value_absence.append(simulated_value)

    x = np.array(simulated_value_absence)
    y = np.array(measured_value_absence)

    try:
        mse = mean_squared_error(y, x)
    except ValueError as e:
        logger.error(e)
        logger.error('-- MSE calculation failed')
        raise e

```

```

return np.sqrt(mse)

def postfn(names, idfOutput, weatherfile, iddfile, choix, thetas):
    """
    our model, run by changing theta within the e+ file

    :param names : noms des parametres a modifier
    :param idfOutput : fichier idf a utiliser
    :param weatherfile : fichier meteo .epw pour les simus
    :param iddfile : fichier Energy+.idd pour les simus
    :param choix : choix (temperature ou conso), sortie equivalente E+
    :param thetas : les parametres
    """

    logger.info('### POSTFN %i###' % os.getpid())
    new_thetas = []
    for theta in thetas :
        if theta < 0 :
            new_thetas.append(abs(theta))
        else : new_thetas.append(theta)
    new_thetas = np.array(new_thetas)
    prefix = splitext(basename(idfOutput))[0]
    idf_file = prefix + "_%i.idf" % os.getpid()
    idf0 = IDF(idfOutput)
    idf0.saveas(idf_file)

    param_set = {}

    for name, thetai in zip(names, new_thetas) :
        param_set[name] = thetai

    idf_file = modif_eplus_abc(idf_file, param_set)

    try :
        df = counter_run(idf_file,
            weatherfile,
            working_dir = "/home/rozenn/Documents/ABCPMC/",
            prefix = splitext(basename(idf_file))[0],
            idd_file= "/usr/local/EnergyPlus-8-4-0/Energy+.idd",
            docker_tag = '')
    except:
        # on relance mais cette fois en sauvegardant les fichiers eplus
        logger.error('----- ERROR POSTFN -----' % os.getpid())
        df = counter_run(idf_file,
            weatherfile,
            working_dir = "/home/rozenn/Documents/ABCPMC/",
            prefix = splitext(basename(idf_file))[0],

```

```

        keep_data = True,
        out_dir = "/home/rozenn/Documents/ABCPMC/eplusFailed/",
        idd_file= "/usr/local/EnergyPlus-8-4-0/Energy+.idd",
        docker_tag = '')
    print "EPLUS FAILED"

    ## TRAITEMENT DES OUTPUTS
    # On lit dans Eppyout.csv la colonne souhaitee
    df = df.set_index(u'Date/Time')
    x = df[choix]
    logger.info('### END POSTFN %i###' % os.getpid())
    return x

```

2.4 Samples et Simulations ABCPMC

on génère les samples à utiliser grâce à une loi normale centrée sur la valeur de notre paramètre et avec un écart type égal à l incertitude on fait tourner le modèle pour chaque valeur du paramètre

```

def generateSamples(means, cov, size, names, y, idfoutput,\
                   iddfile, weatherfile, p, choix):
    """
    :param means : scalar or vector of means
    :param cov : scalar or matrix covariance
    :param size : integer of size of samples
    :param names : names of the parameters
    :param y : measurements
    :param idfOutput : idf file
    :param iddfile : idd file
    :param weatherfile : weatherfilee
    :param p : nombre de processus a utiliser
    :param choix : choix (temperature ou conso), equivalent Et
    """
    means = np.array(means, float)
    mins = np.zeros(means.size)
    maxs = 2*means
    logger.info('-- generate prior')
    prior = TruncatedGaussianPrior(means, cov, mins, maxs)
    # prior = abcpmc.GaussianPrior(mean, sigma)
    logger.info('-- end generate prior')
    logger.info('- generate samples')

    #on initialise notre classe Sampler
    ## on fixe tout sauf theta qui sera modifie dans le abcpmc.Sampler.sampler
    testsamples = abcpmc.Sampler(N=200, Y= y,
                                dist=dist,
                                pool=p,
                                postfn = partial(postfn,

```

```

names,
idfoutput,
weatherfile,
iddfilename,
choix)
)
# décroissance lineaire du seuil de 20C a 0.4C en 300 iterations
#eps = abcpmc.LinearEps(300, 250, 25) # chauffage
eps = abcpmc.LinearEps(300, 5, .2) # temperature
logger.info('- end generate samples')
return prior, eps, testsamples

def ABCPMC(prior, eps, sampler, counter_run, save_posteriori, pools=None):
    """
    lancement de abcpmc

    :param prior : lois a priori
    :param eps : seuils
    :param sampler : fonction generant les echantillons
    :param counter_run : compteur du nombre de simus
    :param save_posteriori : chemin de sauvegarde
    :param pools : pools sauvegardé en cas de bug
    """

    logger.info('-- GENERATE SIMULATION --')
    pools = [] if pools is None else pools
    dico = {'save' : []}
    gen = sampler.sample(prior, eps)
    print "before loop : ", gen

    for pool in gen :
        print "inside loop1"
        iteration = "T: {0}, eps: {1:>.4f}, ratio: {2:>.4f}".format(pool.t,
                                                                    pool.eps,
                                                                    pool.ratio)

        print iteration
        dico['save'].append(iteration)
        for i, (mean, std) in enumerate(zip(np.mean(pool.thetas, axis=0),
                                          np.std(pool.thetas, axis=0))):
            print "inside loop2 : ", i
            theta = u" theta[{0}]: {1:>.4f} +/- {2:>.4f}".format(i, mean, std)
            print theta
            dico['save'].append(theta)
        save = np.array([prior, eps, sampler, pools])
        #np.save('/home/rozzenn/Documents/ABCPMC/backup.npy', save)
        logger.info('-- END GENERATE SIMULATION --')
        df_dico = pd.DataFrame(dico)
        df_dico.to_csv(save_posteriori)

```

```

    pools.append(pool)
    print 'nb total de simu : ', counter_run.counter
    print "end loop1"
df_dico = pd.DataFrame(dico)
df_dico.to_csv(save_posteriori)
return pools, counter_run.counter #, nb_runs

```

3 MAIN

3.1 Définitions du fichier .log

on enregistre les informations dans un fichier .log afin de garder une trace des erreurs éventuelles

```

logger = logging.getLogger()
logger.handlers = []
logger.addHandler(logging.FileHandler(monlog,mode='w'))
logger.setLevel('DEBUG')
#logger.setLevel('ERROR')

```

3.2 Utilisation des décorateurs

Association du décorateur "counter" à la fonction run

On vérifie qu'il n'y a pas de processus zombie

```

counter_run = Counter(run)
counter_run.reset()

```

3.3 Définition des paramètres à étudier

```

Samples = class_Samples.Samples(SourcesPath, idffile)
incertitude, means, names, idfoutput, csv_out, idffile = Samples.sampling_abcpmc()

```

3.4 ABCPMC

3.5 Initialisation et lancement

```

%%time

e=.03

covariance = np.eye(len(means)) * incertitude
counter_run.reset()

new_thetas = []
save_thetas = ref_thetas + 'thetas.csv'
idf_file = modif_eplus(SourcesPath + idffile, SourcesPath + idffileout, e)

df_y = counter_run(idf_file,
    WeatherFile,

```

```

working_dir = "/home/rozenn/Documents/ABCPMC/",
prefix = splitext(basename(idf_filey))[0],
idd_file= "/usr/local/EnergyPlus-8-4-0/Energy+.idd",
docker_tag = '')

df_y = df_y.set_index(u'Date/Time')
y = df_y[choix]
with cPool(30) as p:
    prior, eps, sampler = generateSamples(means,
                                          covariance,
                                          size,
                                          names,
                                          y,
                                          idfoutput,
                                          iddfile,
                                          WeatherFile,
                                          p,
                                          choix)
    pools, nb_simu = ABCPMC(prior, eps, sampler, counter_run, save_posteriori)

```

3.6 Traitement des résultats

3.6.1 Enregistrement de la dernière itération sur le seuil

```

print counter_run.counter
print len(pools)
pool = pools[-1]

df_thetas.empty
for i in range (0,len(pools)):
    new_thetas = ['t={} , eps={}, ratio={}'.format(pool.t, pool.eps, pool.ratio)]
    for j in range (0,len(pools[i].thetas)):
        #print pools[i].thetas[j,0]
        new_thetas.append('{}{}'.format(pools[i].thetas[j,0]))
    df_thetas= pd.DataFrame(new_thetas[1:])
    df_thetas.to_csv('iteration'+str(i)+'.csv', sep = ';')

```

3.6.2 Graphiques

```

import matplotlib.mlab as mlab
import matplotlib.pyplot as plt
from scipy.stats import norm

print "DEBUT DISTANCES"

print (np.mean(abs(pool.thetas[:,0])), np.std(abs(pool.thetas[:,0])))

pool = pools[-1]

```

```

for i, name in enumerate(names) :
    posteriori = np.array(abs(pool.thetas[:,i]))
    fig, ax1 = plt.subplots()
    ax1.set_ylabel('Frequence', fontsize = '14')
    if 'thickness' in name :
        ax1.set_xlabel('Epaisseur d\'isolant [cm]', fontsize = '14')
        posteriori = posteriori *100
    else :
        if 'conductivity' in name :
            ax1.set_xlabel('Conductivite de l\'isolant [W/(m.K)]', fontsize = '14')
        else :
            ax1.set_xlabel(name, fontsize = '14')
    ax1.tick_params('both', labelsize=14)
    sns.distplot(posteriori,fit = norm, kde=False, ax = ax1)
    fig.savefig(ResultsPath + ref_results + name.replace('/', '_') + '.png')
    fig.savefig(ResultsPath + ref_results + name.replace('/', '_') + '.pdf')
sns.plt.show()

```

Class_IDF

September 13, 2017

Préambule

class IdfFile(object):

precise les chemins de travail : input, output, localisation E+, etc

ouvre le fichier idf et recupere les elements modifiables

class SelectionParam(IdfFile):

recupere les proprietes modifiables des elements precedents

demande a l'utilisateur de selectionner les parametres de travail pour l'AS de Morris

Code

```
#!/usr/bin/python2.7
#-*- coding: utf-8 -*-

__author__ = 'rozenn.josse-buret@g2elab.grenoble-inp.fr'

import sys
from eppy.modeleditor import IDF
import pandas as pd

class IdfFile(object):
    def __init__(self, MainPath, idffile):
        print "##### Recherche fichier idf #####"
        self.EnergyPlusPath = MainPath
        self.outputPath = MainPath
        self.idfInput = self.EnergyPlusPath + idffile
        self.idfOutput = self.EnergyPlusPath + "Eppyout.idf"
        from shutil import copyfile; copyfile(self.idfInput, self.idfOutput)
        self.fsimulresults = self.EnergyPlusPath + "Eppyout.csv"
        self.htmlfile = self.EnergyPlusPath + "EppyoutTable.html"
        self.WeatherFileName = "FR-Carpentras-75860TM2"
        sys.path.append(MainPath)
```

```

sys.path.append(self.EnergyPlusPath)
self.iddfile = "/usr/local/EnergyPlus-8-4-0/Energy+.idd"

try:
    IDF.setiddname(self.iddfile)
except Exception:
    print Exception

def openIDF(self):
    idf1 = IDF(self.idfInput)
    print "L'analyse est effectuee sur : ", self.idfInput, "\n"

    # on recupere toutes les caracteristiques des materiaux
    materials = idf1.idfobjects["MATERIAL"]
    windows = idf1.idfobjects["WINDOWMATERIAL:SIMPLEGLAZINGSYSTEM"]
    people = idf1.idfobjects["PEOPLE"]
    electric equip = idf1.idfobjects["ELECTRICEQUIPMENT"]
    infiltration = idf1.idfobjects["ZONEINFILTRATION:DESIGNFLOWRATE"]
    ventilation = idf1.idfobjects["ZONEVENTILATION:DESIGNFLOWRATE"]

    return idf1, materials, windows, people, electric equip, infiltration, ventilation

class SelectionParam(IdfFile):
    def __init__(self, MainPath, iddfile):
        super(SelectionParam, self).__init__(MainPath, iddfile)
        ## on recupere les elements modifiables du fichier idf
        self.file = self.openIDF()
        self.idf1 = self.file[0]
        self.materials = self.file[1]
        self.windows = self.file[2]
        self.people = self.file[3]
        self.elec = self.file[4]
        self.infiltration = self.file[5]
        self.ventilation = self.file[6]

        ## creation de listes pour recuperer toutes informations disponibles
        ## concernant les materiaux
        self.name_mat = []
        self.roughness = []
        self.thickness = []
        self.conductivity = []
        self.density = []
        self.specific_heat = []
        self.thermal_absorbance = []
        self.solar_absorbance = []
        self.visible_absorbance = []

        ## creation de listes pour recuperer toutes informations disponibles

```

```

    ## concernant les fenetres
    self.name_win = []
    self.ufactor = []
    self.solar_heat_gain = []
    self.visible_transmittance = []

    ## creation de listes pour recupere toutes les informations disponibles
    ## concernant les scenarios
    self.name_people = []
    self.calculation_method = []

    self.name_infil = []
    self.infil_design_flow_rate_calculation_method = []

    self.name_ventil = []
    self.ventil_design_flow_rate_calculation_method = []

    self.name_elec = []
    self.design_level_calculation_method = []

def selection_param(self):
    print "#### RECUPERATION DES CARACTERISTIQUES DES MATERIAUX ####\n"
    for material in self.materials:
        self.name_mat.append(material.Name)
        self.thickness.append(material.Thickness)
        self.conductivity.append(material.Conductivity)
        self.density.append(material.Density)
        self.specific_heat.append(material.Specific_Heat)
        self.thermal_absorbance.append(material.Thermal_Absorptance)
        self.solar_absorbance.append(material.Solar_Absorptance)
        self.visible_absorbance.append(material.Visible_Absorptance)

    dict_materials = {"Name": self.name_mat,"Thickness": self.thickness,
                     "Conductivity": self.conductivity, "Density": self.density, \
                     "Specific_Heat": self.specific_heat}
    df_materials = pd.DataFrame(dict_materials)
    cols = df_materials.columns.tolist()
    # on ordonne les colonnes du dataframe
    cols = cols[2:3] + cols [4:] + cols [0:1] + cols [3:4] + cols [1:2]
    df_materials = df_materials[cols]
    print("La liste des materiaux est la suivante:\n "
          "{}".format(df_materials))
    print "\n" \
          "Il y a {} materiaux \n".format(len(self.materials))

    print "#### CHOIX DES MATERIAUX A ETUDIER #### \n"
    list_materials = []
    str_ask =raw_input("Voulez vous etudier tous les materiaux (oui = 1, non = 0)? : ")

```

```

ask = int(str_ask)
if ask == 1:
    list_materials = list(df_materials["Name"])
else:
    if ask == 0:
        exit_boucle = 1
        while exit_boucle == 1:
            if len(list_materials) == len(df_materials.index):
                print "tous les materiaux ont ete selectionnes"
                exit_boucle = 1

            str_index = raw_input("Donnez l'index du materiau a etudier : ")
            index = int(str_index)
            if df_materials["Name"][index] in list_materials:
                print 'Ce materiau a deja ete selectionne ! '
            else : list_materials.append(df_materials["Name"][index])
            str_exit_boucle = raw_input("Voulez vous etudier d'autres materiaux \
                (oui = 1, non = 0) : ")
            exit_boucle = int(str_exit_boucle)
        else :
            print 'Erreur :'
            str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                tous les materiaux, 0 sinon : ")
            ask = int(str_ask)
print ""

print "#### CHOIX DES CARACTERISTIQUES DES MATERIAUX A ETUDIER ####\n"
print "Les caracteristiques des materiaux sont les suivantes :"
for i in range (0,len(df_materials.columns[1:])):
    print "{} \t {}".format(i, df_materials.columns[i+1])
print ""
str_ask = raw_input("Voulez vous etudier toutes les caracteristiques \
    des materiaux (oui = 1, non = 0) : ")
ask = int(str_ask)
list_caract_mat = []

if ask == 1:
    list_caract_mat = list(df_materials.columns[1:])
else:
    if ask == 0:
        exit_boucle = 1
        while exit_boucle == 1:
            if len(list_caract_mat) == len(df_materials.columns[1:]):
                print "toutes les caracteristiques ont ete selectionnees"
                exit_boucle = 1
            str_index = raw_input("Donnez l'index de la caracteristique a etudier :")
            index = int(str_index)
            if df_materials.columns[index+1] in list_caract_mat:

```

```

        print 'Cette caracteristique a deja ete selectionnee ! '
    else : list_caract_mat.append(df_materials.columns[index+1])
    str_exit_boucle = raw_input("Voulez vous etudier d'autres \
                                caracteristiques (oui = 1, non = 0) : ")
    exit_boucle = int(str_exit_boucle)
else :
    print 'Erreur :'
    str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                        toutes les caracteristiques, 0 sinon : ")
    ask = int(str_ask)
print ""

print "#### RECUPERATION DES CARACTERISTIQUES DES FENETRES ####\n"
for window in self.windows :
    self.name_win.append(window.Name)
    self.ufactor.append(window.UFactor)
    self.solar_heat_gain.append(window.Solar_Heat_Gain_Coefficient)
    self.visible_transmittance.append(window.Visible_Transmittance)

dict_windows = {"Name": self.name_win, "UFactor": self.ufactor, \
                "Solar_Heat_Gain_Coefficient": self.solar_heat_gain,\
                "Visible_Transmittance": self.visible_transmittance}

df_windows = pd.DataFrame(dict_windows)
print("La liste des fenetres est la suivante:\n "
      "{}".format(df_windows))
print "\n" \
      "Il y a {} fenetres\n".format(len(self.windows))

print "#### CHOIX DES FENETRES A ETUDIER ####\n"
list_windows = []
list_caract_win = []
str_ask0 = raw_input("Voulez vous etudier les fenetres (oui = 1, non = 0)? : ")
ask0 = int(str_ask0)
if ask0 == 1 :
    str_ask = raw_input("Voulez vous etudier \
                        toutes les fenetres (oui = 1, non = 0)? : ")
    ask = int(str_ask)
    if ask == 1:
        list_windows = list(df_windows["Name"])
    else:
        if ask == 0:
            exit_boucle = 1
            while exit_boucle == 1:
                if len(list_windows) == len(df_windows.index):
                    print "toutes les fenetres ont ete selectionnees"
                    exit_boucle = 1
                str_index = raw_input("Donnez l'index de la fenetre a etudier : ")

```

```

        index = int(str_index)
        if df_windows["Name"][index] in list_windows:
            print 'Cette fenetre a deja ete selectionnee ! '
        else : list_windows.append(df_windows["Name"][index])

        str_exit_boucle = raw_input("Voulez vous etudier \
                                     d'autres fenetres (oui = 1, non = 0) :")
        exit_boucle = int(str_exit_boucle)
    else:
        print 'Erreur :'
        str_ask = raw_input("Veillez entrer 1 si vous voulez etudier \
                             tous les materiaux, 0 sinon : ")
        ask = int(str_ask)
print ""

print "#### CHOIX DES CARACTERISTIQUES DES FENETRES A ETUDIER ####\n"
print "Les caracteristiques des fenetres sont les suivantes :"
for i in range (0,len(df_windows.columns[1:])):
    print "{} \t {}".format(i, df_windows.columns[i+1])
print ""

str_ask = raw_input("Voulez vous etudier toutes les caracteristiques \
                    des fenetres (oui = 1, non = 0) : ")
ask = int(str_ask)
if ask == 1:
    list_caract_win = list(df_windows.columns[1:])
else:
    if ask == 0:
        exit_boucle = 1
        while exit_boucle == 1:
            if len(list_caract_win) == len(df_windows.columns[1:]):
                print "toutes les caracteristiques ont ete selectionnees"
                exit_boucle = 1
            str_index = raw_input("Donnez l'index de la caracteristique a etudier :")
            index =int(str_index)
            if df_windows.columns[index+1] in list_caract_win:
                print 'Cette caracteristique a deja ete selectionnee ! '
            else : list_caract_win.append(df_windows.columns[index+1])
            str_exit_boucle = raw_input("Voulez vous etudier d'autres \
                                        caracteristiques (oui = 1, non = 0) : ")
            exit_boucle = int(str_exit_boucle)
        else :
            print 'Erreur :'
            str_ask = raw_input("Veillez entrer 1 si vous voulez etudier \
                                toutes les caracteristiques, 0 sinon : ")
            ask = int(str_ask)
print ""

```

```

else :
    list_windows = []
    list_caract_win = []

print "#### RECUPERATION DES 'SCENARIOS' DE PRESENCE MODIFIABLES ####\n"
for people in self.people :
    self.name_people.append(people.Name)
    self.calculation_method.append(people.Number_of_People_Calculation_Method)

dict_people = {"Name": self.name_people, "Calculation_Method": self.calculation_method}

df_people = pd.DataFrame(dict_people)
print("La liste des scenarios de presence est la suivante:\n "
      "{}".format(df_people))
print "\n" \
      "Il y a {} scenarios de presence\n".format(len(self.people))

print "#### CHOIX DES SCENARIOS DE PRESENCE A ETUDIER ####\n"
list_people = []
str_ask0 = raw_input("Voulez vous etudier les scenarios \
                    de presence (oui = 1, non = 0)? : ")

ask0 = int (str_ask0)
if ask0 == 1 :
    str_ask = raw_input("Voulez vous etudier tous les \
                    scenarios de presence (oui = 1, non = 0)? : ")

    ask = int(str_ask)
    if ask == 1:
        list_people = list(df_people["Name"])
    else:
        if ask == 0:
            exit_boucle = 1
            while exit_boucle == 1:
                if len(list_people) == len(df_people.index):
                    print "tous les scenarios de presence ont ete selectionnes"
                    exit_boucle = 1
                str_index = raw_input("Donnez l'index du scenario a etudier : ")
                index = int(str_index)
                if df_people["Name"][index] in list_people:
                    print 'Ce scenario a deja ete selectionne ! '
                else : list_people.append(df_people["Name"][index])

                str_exit_boucle = raw_input("Voulez vous etudier \
                    d'autres scenarios (oui = 1, non = 0) :")
                exit_boucle = int(str_exit_boucle)
        else:
            print 'Erreur :'
            str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                    tous les scenarios, 0 sinon : ")

```

```

        ask = int(str_ask)
    print ""
else :
    list_people = []

print "#### RECUPERATION DES 'SCENARIOS' D'EQUIPEMENTS ELECTRIQUES MODIFIABLES ####\n"
for elec in self.elec :
    self.name_elec.append(elec.Name)
    self.design_level_calculation_method.append(elec.Design_Level_Calculation_Method)

dict_elec = {"Name": self.name_elec, \
            "Calculation_Method": self.design_level_calculation_method}

df_elec = pd.DataFrame(dict_elec)
print("La liste des scenarios des equipements electriques est la suivante:\n "
      "{}".format(df_elec))
print "\n" \
      "Il y a {} scenarios des equipements electriques\n".format(len(self.elec))

print "#### CHOIX DES SCENARIOS D'EQUIPEMENTS ELECTRIQUES A ETUDIER ####\n"
list_elec = []
str_ask0 = raw_input("Voulez vous etudier les scenarios \
                    d'equipements electriques (oui = 1, non = 0)? : ")
ask0 = int(str_ask0)
if ask0 == 1 :
    str_ask = raw_input("Voulez vous etudier tous les scenarios \
                    des equipements electriques (oui = 1, non = 0)? : ")
    ask = int(str_ask)
    if ask == 1:
        list_elec = list(df_elec["Name"])
    else:
        if ask == 0:
            exit_boucle = 1
            while exit_boucle == 1:
                if len(list_elec) == len(df_elec.index):
                    print "tous les scenarios des equipements \
                            electriques ont ete selectionnes"
                    exit_boucle = 1
                str_index = raw_input("Donnez l'index du scenario a etudier : ")
                index = int(str_index)
                if df_elec["Name"][index] in list_elec:
                    print 'Ce scenario a deja ete selectionne ! '
                else : list_elec.append(df_elec["Name"][index])
                str_exit_boucle = raw_input("Voulez vous etudier \
                    d'autres scenarios (oui = 1, non = 0) :")
                exit_boucle = int(str_exit_boucle)
        else:
            print 'Erreur :'

```

```

        str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                            tous les scenarios, 0 sinon : ")
        ask = int(str_ask)
    print ""
else :
    list_elec = []

print "#### RECUPERATION DES 'SCENARIOS' D'INFILTRATIONS MODIFIABLES ####\n"
for infil in self.infiltration :
    self.name_infil.append(infil.Name)
    self.infil_design_flow_rate_calculation_method.append(
        infil.Design_Flow_Rate_Calculation_Method)

dict_infil = {"Name": self.name_infil, \
              "Calculation_Method": self.infil_design_flow_rate_calculation_method}

df_infil = pd.DataFrame(dict_infil)
print("La liste des scenarios d'infiltrations est la suivante:\n "
      "{}".format(df_infil))
print "\n \
      "Il y a {} scenarios d'infiltrations\n".format(len(self.infiltration))

print "#### CHOIX DES SCENARIOS D'INFILTRATIONS A ETUDIER ####\n"
list_infil =[]

str_ask0 = raw_input("Voulez vous etudier \
                    les scenarios d'infiltrations (oui = 1, non = 0)? : ")
ask0 = int(str_ask0)
if ask0 == 1 :
    str_ask = raw_input("Voulez vous etudier tous les etudier (oui = 1, non = 0)? : ")
    ask = int(str_ask)
    if ask == 1:
        list_infil = list(df_infil["Name"])
    else:
        if ask == 0:
            exit_boucle = 1
            while exit_boucle == 1:
                if len(list_infil) == len(df_infil.index):
                    print "toutes les scenarios d'infiltrations ont ete selectionnes"
                    exit_boucle = 1
                str_index = raw_input("Donnez l'index du scenario a etudier : ")
                index = int(str_index)
                if df_infil["Name"][index] in list_infil:
                    print 'Ce scenario a deja ete selectionne ! '
                else : list_infil.append(df_infil["Name"][index])
                str_exit_boucle = raw_input("Voulez vous etudier \
                    d'autres scenarios (oui = 1, non = 0) :")
                exit_boucle = int(str_exit_boucle)

```

```

        else:
            print 'Erreur :'
            str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                                tous les scenarios, 0 sinon : ")

            ask= int(str_ask)
        print ""
    else :
        list_infil = []

print "#### RECUPERATION DES 'SCENARIOS' DE VENTILATION MODIFIABLES ####\n"
for ventil in self.ventilation :
    self.name_ventil.append(ventil.Name)
    self.ventil_design_flow_rate_calculation_method.append(
        ventil.Design_Flow_Rate_Calculation_Method)

dict_ventil = {"Name": self.name_ventil, \
               "Calculation_Method": self.ventil_design_flow_rate_calculation_method}

df_ventil = pd.DataFrame(dict_ventil)
print("La liste des scenarios de ventilation est la suivante:\n "
      "{}".format(df_ventil))
print "\n" \
      "Il y a {} scenarios de ventilation\n".format(len(self.ventilation))

print "#### CHOIX DES SCENARIOS DE VENTILATION A ETUDIER ####\n"
list_ventil =[]

str_ask0 = raw_input("Voulez vous etudier les scenarios \
                    de ventilation (oui = 1, non = 0)? : ")
ask0 = int(str_ask0)
if ask0 == 1 :
    str_ask = raw_input("Voulez vous etudier tous les etudier (oui = 1, non = 0)? : ")
    ask = int(str_ask)
    if ask == 1:
        list_ventil = list(df_ventil["Name"])
    else:
        if ask == 0:
            exit_boucle = 1
            while exit_boucle == 1:
                if len(list_ventil) == len(df_ventil.index):
                    print "toutes les scenarios de ventilation ont ete selectionnes"
                    exit_boucle = 1
                str_index = raw_input("Donnez l'index du scenario a etudier : ")
                index = int(str_index)
                if df_ventil["Name"][index] in list_ventil:
                    print 'Ce scenario a deja ete selectionne ! '
                else : list_ventil.append(df_ventil["Name"][index])
                str_exit_boucle = raw_input("Voulez vous etudier \

```

```

                                d'autres scenarios (oui = 1, non = 0) :)")
        exit_boucle = int(str_exit_boucle)
    else:
        print 'Erreur :'
        str_ask = raw_input("Veuillez entrer 1 si vous voulez etudier \
                                tous les scenarios, 0 sinon : ")
        ask = int(str_ask)
    print ""
else :
    list_ventil = []

return list_materials, list_caract_mat, \
        list_windows, list_caract_win, \
        list_people, list_elec, list_infil, list_ventil

```

Class_Samples

September 13, 2017

1 Table of Contents

class Samples(SelectionParam) :
1/MORRIS

a/ cree les limites de variation pour l'AS de Morris des parametres
selectionnés dans SelectionParam

b/ genere le probleme d'etude pour l'AS de Morris

c/ genere les samples pour effectuer les simulations pour l'AS de Morris

2/ABCPMC

genere les samples pour effectuer les simulations pour l'abcpmc

```
#!/usr/bin/python2.7  
#-*- coding: utf-8 -*-
```

```
__author__ = 'rozenn.josse-buret@g2elab.grenoble-inp.fr'
```

```
import SALib.sample.morris as moS  
import pandas as pd  
from class_IDF import SelectionParam  
import numpy as np
```

```
class Samples(SelectionParam) :  
    def __init__(self, MainPath, idffile):  
        super(Samples,self).__init__(MainPath, idffile)  
  
        # recuperation des parametres selectionnes dans SelectionParam()  
        lists = self.selection_param()  
        self.list_mat, self.list_caract_mat, \  
        self.list_win, self.list_caract_win, \  
        self.list_people, self.list_elec, \  
        self.list_infil, self.list_ventil = lists
```

```

def sampling_abcPMC(self):
    param_names = []
    uncertainties = []
    means = []
    ### ETABLISSEMENT DES INCERTITUDES POUR CHAQUE PARAMETRE ###
    for caract in self.list_caract_mat:
        for name in self.list_mat:
            for material in self.materials:
                if name in material.Name:
                    if caract == "Thickness":
                        #### Conditions sur l'epaisseur des materiaux [m] ####
                        # print "{} : {}".format(material.Name, material.Thickness)
                        uncertainties.append(0.9* material.Thickness)
                        means.append(material.Thickness)
                        param_names.append("{} thickness (m)".format(material.Name))

                    if caract == "Conductivity":
                        #### Conditions sur la conductivite [W/m.K] ####
                        uncertainties.append(0.1 * material.Conductivity)
                        means.append(material.Conductivity)
                        param_names.append("{} conductivity (W/m.K)".format(material.Name))

                    if caract == "Specific_Heat":
                        #### Conditions sur la chaleur specifique [J/kg.K] ####
                        uncertainties.append(0.1 * material.Specific_Heat)
                        means.append(material.Specific_Heat)
                        param_names.append("{} specific heat \
(J/kg.K)".format(material.Name))

                    if caract == "Density":
                        #### Conditions sur la densite [kg/m3] ####
                        uncertainties.append(0.1 * material.Density)
                        means.append(material.Density)
                        param_names.append("{} density \
(kg/m3)".format(material.Name))

    for caract in self.list_caract_win:
        for name in self.list_win:
            for window in self.windows:
                if name in window.Name:
                    if caract == "Solar_Heat_Gain_Coefficient":
                        #### Conditions sur le facteur solaire (-) ####
                        solar_heat_gain = float(window.Solar_Heat_Gain_Coefficient)
                        uncertainties.append(0.1 * solar_heat_gain)
                        means.append(solar_heat_gain)
                        param_names.append("{} Solar_Heat_Gain_Coefficient (-)".format(
                            window.Name)
                    )

```

```

    if caract == "UFactor":
        ##### Conditions sur le facteur U (W/m2.K) #####
        # print "{} : {}".format(window.Name, window.UFactor)
        ufactor = float(window.UFactor)
        uncertainties.append(0.1 * ufactor)
        means.append(ufactor)
        param_names.append("{} UFactor (W/m2.K)".format(window.Name))

    if caract == "Visible_Transmittance":
        # print "{} : {}".format(window.Name, window.Visible_Transmittance)
        transmittance = float(window.Visible_Transmittance)
        uncertainties.append(0.1 * transmittance)
        means.append(transmittance)
        param_names.append("{} Visible Transmittance".format(window.Name))

for name in self.list_people:
    for person in self.people:
        if name in person.Name:
            ##### Conditions sur le nb de personnes #####
            if person.Number_of_People_Calculation_Method == "People/Area":
                nb_of_people = float(person.People_per_Zone_Floor_Area)
                param_names.append("{} People/Area".format(person.Name))
            if person.Number_of_People_Calculation_Method == "People":
                nb_of_people = float(person.Number_of_People)
                param_names.append("{} People".format(person.Name))
            uncertainties.append(0.1 * nb_of_people)
            means.append(nb_of_people)

for name in self.list_elec:
    for elec in self.elec:
        if name in elec.Name:
            ##### Conditions sur les apports internes #####
            if elec.Design_Level_Calculation_Method == "Watts/Area":
                power = float(elec.Watts_per_Zone_Floor_Area)
                param_names.append("{} Watts/Area (W/m2)".format(elec.Name))
            if elec.Design_Level_Calculation_Method == "EquipmentLevel":
                power = float(elec.Design_Level)
                param_names.append("{} EquipmentLevel (W)".format(elec.Name))
            uncertainties.append(0.1 * power)
            means.append(power)

for name in self.list_infil:
    for infil in self.infiltration:
        if name in infil.Name:
            ##### Conditions sur les infiltrations #####
            if infil.Design_Flow_Rate_Calculation_Method == "Flow/Zone":
                debit = float(infil.Design_Flow_Rate)

```

```

        param_names.append("{} flow rate (m3/s)".format(infil.Name))
    if infil.Design_Flow_Rate_Calculation_Method == "AirChanges/Hour":
        debit = float(infil.Air_Changes_per_Hour)
        param_names.append("{} air changes (1/hr)".format(infil.Name))
    uncertainties.append(0.1 * debit)
    means.append(debit)

for name in self.list_ventil:
    for ventil in self.ventilation:
        if name in ventil.Name:
            ##### Conditions sur les infiltrations #####
            if ventil.Design_Flow_Rate_Calculation_Method == "Flow/Zone":
                debit = float(ventil.Design_Flow_Rate)
                param_names.append("{} flow rate (m3/s)".format(ventil.Name))
            if ventil.Design_Flow_Rate_Calculation_Method == "AirChanges/Hour":
                debit = float(ventil.Air_Changes_per_Hour)
                param_names.append("{} air changes (1/hr)".format(ventil.Name))
            uncertainties.append(0.1 * debit)
            means.append(debit)

# on remplace les espaces par des underscores pour faciliter
# le traitement csv par la suite
for i in range(0, len(param_names)):
    param_names[i] = param_names[i].replace(' ', '_')

return np.array(uncertainties), np.array(means), param_names, \
        self.idfOutput, self.fsimulresults, self.iddfile

def sampling_analyzeMorris(self):
    # initialisation des listes pour le sampling de Morris
    inf_bound = []
    sup_bound = []
    parameters_names = []
    elementsList = []
    ### ETABLISSEMENT DES LIMITES POUR CHAQUE PARAMETRE ###
    for caract in self.list_caract_mat:
        for name in self.list_mat :
            for material in self.materials:
                if name in material.Name:
                    if caract == "Thickness":
                        ##### Conditions sur l'epaisseur des materiaux [m] #####
                        #print "{} : {}".format(material.Name, material.Thickness)
                        inf_bound.append(material.Thickness - 0.1*material.Thickness)
                        sup_bound.append(material.Thickness + 0.1*material.Thickness)
                        parameters_names.append("{} thickness (m)".format(material.Name))
                        elementsList.append(material.Thickness)

                    if caract == "Conductivity":

```

```

##### Conditions sur la conductivite [W/m.K] #####
#print "{} : {}".format(material.Name, material.Conductivity)
inf_bound.append(material.Conductivity - 0.1*material.Conductivity)
sup_bound.append(material.Conductivity + 0.1*material.Conductivity)
parameters_names.append("{} conductivity (W/m.K)".format(
                                                                    material.Name)
)
elementsList.append(material.Conductivity)

if caract == "Specific_Heat":
##### Conditions sur la chaleur specifique [J/kg.K] #####
#print "{} : {}".format(material.Name, material.Specific_Heat)
inf_bound.append(
    material.Specific_Heat - 0.1*material.Specific_Heat
)
sup_bound.append(
    material.Specific_Heat + 0.1*material.Specific_Heat
)
parameters_names.append("{} specific heat (J/kg.K)".format(
                                                                    material.Name
)
)
elementsList.append(material.Specific_Heat)

if caract == "Density":
##### Conditions sur la densite [kg/m3] #####
#print "{} : {}".format(material.Name, material.Density)
inf_bound.append(material.Density - 0.1*material.Density)
sup_bound.append(material.Density + 0.1*material.Density)
parameters_names.append("{} density (kg/m3)".format(material.Name))
elementsList.append(material.Density)

for caract in self.list_caract_win:
    for name in self.list_win :
        for window in self.windows:
            if name in window.Name:
                if caract == "Solar_Heat_Gain_Coefficient":
##### Conditions sur le facteur solaire (-) #####
solar_heat_gain = float(window.Solar_Heat_Gain_Coefficient)
inf_bound.append(solar_heat_gain - 0.1*solar_heat_gain)
sup_bound.append(solar_heat_gain + 0.1*solar_heat_gain)
parameters_names.append("{} Solar_Heat_Gain_Coefficient (-)\
                                                                    ".format(window.Name))
elementsList.append(window.Solar_Heat_Gain_Coefficient)

                if caract == "UFactor":
##### Conditions sur le facteur U (W/m2.K) #####
#print "{} : {}".format(window.Name, window.UFactor)

```

```

        ufactor = float(window.UFactor)
        inf_bound.append(ufactor - 0.1*ufactor)
        sup_bound.append(ufactor + 0.1*ufactor)
        parameters_names.append("{} UFactor (W/m2.K)".format(window.Name))
        elementsList.append(window.UFactor)

    if caract == "Visible_Transmittance":
        #print "{} : {}".format(window.Name, window.Visible_Transmittance)
        transmittance = float(window.Visible_Transmittance)
        inf_bound.append(transmittance - 0.1*transmittance)
        sup_bound.append(transmittance + 0.1*transmittance)
        parameters_names.append("{} Visible Transmittance\
                                ".format(window.Name))
        elementsList.append(window.Visible_Transmittance)

for name in self.list_people :
    for person in self.people:
        if name in person.Name:
            ##### Conditions sur le nb de personnes #####
            if person.Number_of_People_Calculation_Method == "People/Area":
                nb_of_people = float(person.People_per_Zone_Floor_Area)
                parameters_names.append("{} People/Area".format(person.Name))
                elementsList.append(person.People_per_Zone_Floor_Area)
            if person.Number_of_People_Calculation_Method == "People":
                nb_of_people = float(person.Number_of_People)
                parameters_names.append("{} People".format(person.Name))
                elementsList.append(person.Number_of_People)
            inf_bound.append(nb_of_people - 0.2*nb_of_people)
            sup_bound.append(nb_of_people + 0.2*nb_of_people)

for name in self.list_elec :
    for elec in self.elec:
        if name in elec.Name:
            ##### Conditions sur les apports internes #####
            if elec.Design_Level_Calculation_Method == "Watts/Area":
                power = float(elec.Watts_per_Zone_Floor_Area)
                parameters_names.append("{} Watts/Area (W/m2)".format(elec.Name))
                elementsList.append(elec.Watts_per_Zone_Floor_Area)
            if elec.Design_Level_Calculation_Method == "EquipmentLevel":
                parameters_names.append("{} EquipmentLevel (W)".format(elec.Name))
                elementsList.append(elec.Design_Level)
            inf_bound.append(power - 0.2*power)
            sup_bound.append(power + 0.2*power)

for name in self.list_infil :
    for infil in self.infiltration:
        if name in infil.Name:
            ##### Conditions sur les infiltrations #####

```

```

if infil.Design_Flow_Rate_Calculation_Method == "Flow/Zone":
    debit = float(infil.Design_Flow_Rate)
    parameters_names.append("{} flow rate (m3/s)".format(infil.Name))
    elementsList.append(infil.Design_Flow_Rate)
if infil.Design_Flow_Rate_Calculation_Method == "AirChanges/Hour":
    debit = float(infil.Air_Changes_per_Hour)
    parameters_names.append("{} air changes (1/hr)".format(infil.Name))
    elementsList.append(infil.Air_Changes_per_Hour)
inf_bound.append(debit - 0.1*debit)
sup_bound.append(debit + 0.1*debit)

for name in self.list_ventil :
    for ventil in self.ventilation:
        if name in ventil.Name:
            #### Conditions sur les infiltrations ####
            if ventil.Design_Flow_Rate_Calculation_Method == "Flow/Zone":
                debit = float(ventil.Design_Flow_Rate)
                parameters_names.append("{} flow rate (m3/s)".format(ventil.Name))
                elementsList.append(ventil.Design_Flow_Rate)
            if ventil.Design_Flow_Rate_Calculation_Method == "AirChanges/Hour":
                debit = float(ventil.Air_Changes_per_Hour)
                parameters_names.append("{} air changes (1/hr)".format(ventil.Name))
                elementsList.append(ventil.Air_Changes_per_Hour)
            inf_bound.append(debit - 0.1*debit)
            sup_bound.append(debit + 0.1*debit)

# on remplace les espaces par des underscores
# pour faciliter le traitement csv par la suite
for i in range (0, len(parameters_names)):
    parameters_names[i] = parameters_names[i].replace(' ', '_')

### on cree les couples de bornes pour la definition du probleme d'analyse
bounds = []
for i in range (0,len(inf_bound)):
    bounds.append([inf_bound[i], sup_bound[i]])

print "attention nombre de samples doit etre un multiple de {}+1 \n\
      ".format(len(parameters_names))
N = int(raw_input("Donner le nombre de sample a generer : "))

##### Analyse de sensibilite - Methode de Morris #####
problem = {
    'num_vars': len(parameters_names),
    'names': parameters_names,
    'groups': None,
    'bounds': bounds
}

```

```
param_morris = problem
param_morris['samples'] = N

X = moS.sample(problem, N, 8, 2, optimal_trajectories=None)

df_X = pd.DataFrame(X)
df_X.columns = parameters_names

return problem, N, X, df_X
```


Annexe C

Maison POS-A



Table des matières

1	Description de la maison	206
2	Modèle thermique et résultats attendus	207
2.1	Caractéristiques thermiques	207
2.2	Performances énergétiques attendues	208
D	Requêtes http pour les webserveurs	211

1 Description de la maison



Photo maison construite Localisation Nb d'occupants Description des pièces

La Maison Pos-A [C.1], de Mas Provence, une maison individuelle située à CHATEAURENARD en zone H3, dans une zone résidentielle. Elle a été conçue pour répondre aux besoins d'une famille de 5 personnes (1 couple actif et 3 enfants). La construction s'est terminée en 2015 et est habitée depuis.



Figure C.1 – Maison Pos-A.

Elle est composée de 4 demi-niveaux. Au rez-de-chaussée se trouvent le garage, l'entrée et trois chambres (celles des enfants) avec salles d'eau. Au 1er étage se trouve un bureau pouvant servir de séjour ainsi que le cellier où se situe l'armoire électrique. Au 2ème étage, on retrouve l'espace de vie principal avec la salle à manger et la cuisine donnant accès à la terrasse. Enfin, au 3ème étage, il y a la suite parentale avec chambre, dressing et salle d'eau.

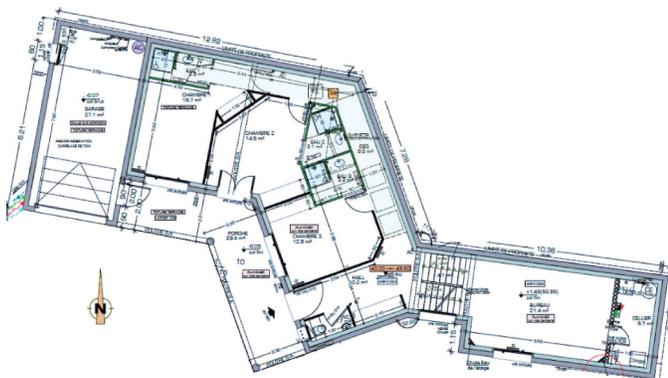


Figure C.2 – Maison POS-A - Plan RDC.

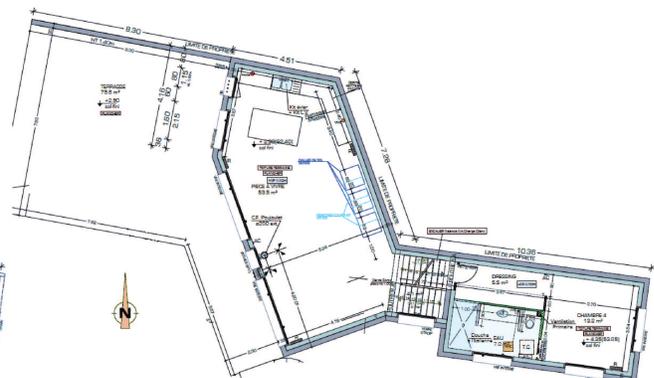


Figure C.3 – Maison POS-A - Plan R+1.

2 Modèle thermique et résultats attendus △

2.1 Caractéristiques thermiques

Nous répertorions ici les principales caractéristiques thermiques qui ont permis d'aboutir à une estimation des consommations de la maison POS-A. Ces informations sont principalement issues du livrable 5.1.a (Hadj Saïd et al. [50]) du projet COMEPOS, dans lequel ces travaux de thèse se sont déroulés.

Les conditions aux limites et la composition de l'enveloppe sont décrites ici. Les murs extérieurs présentent une résistance thermique totale des isolants de 6,12 (m².K)/W (U= 0,16 W/(m²_{paroi}.K)). Le détail de leur composition (de l'extérieur vers l'intérieur) est récapitulé dans le tableau C.1. Les fenêtres sont en double vitrage avec un intercalaire de 15 millimètres d'argon.

Matériau	Épaisseur [cm]	Densité [kg/m ³]	Conductivité [W/(m.K)]	Chaleur spécifique [J/(kg.K)]
Enduit mono-couche	1	1100	0,23	1000
Béton d'argile expansée	20	600	0,182	1000
Laine minérale	18	40	0,032	1030
Plaque de plâtre	1,3	1200	0,41	1000

Table C.1 – Caractéristiques des murs extérieurs (de l'extérieur vers l'intérieur).

Les ponts thermiques considérés sont résumés dans le tableau C.2 ci-dessous.

	Ponts thermiques transversale aux poutrelles	Ponts thermiques longitudinale aux poutrelles
Dalle terrasse sur mur façade à l'étage (20+5 poutrelles + hourdis)	0,20 W/(m.K)	0,13 W/(m.K)
Dalle sur VS (16+4 poutrelles + hourdis)	0,24 W/(m.K)	0,13 W/(m.K)
Dalle intermédiaire (20+5 poutrelles + hourdis)	0,31 W/(m.K)	0,18 W/(m.K)

Table C.2 – Ponts thermiques considérés

En ce qui concernent les infiltrations, elles s'élèvent à 0,6 m³/h/m² selon le constructeur Mas Provence.

Les figures C.4 et C.5 présentent le modèle thermique, modélisé sous Sketchup. La figure C.6 localise les matériaux sur le modèle 3D réalisé sous Sketchup.

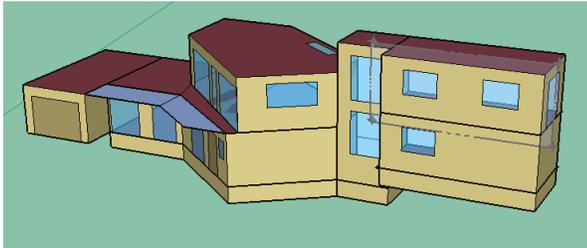


Figure C.4 – Maison POS-A - Modèle Sketchup.

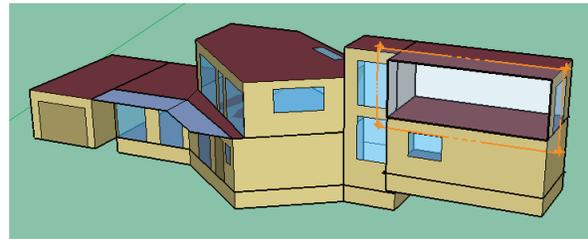


Figure C.5 – Maison POS-A - Modèle Sketchup : coupe chambre parents.

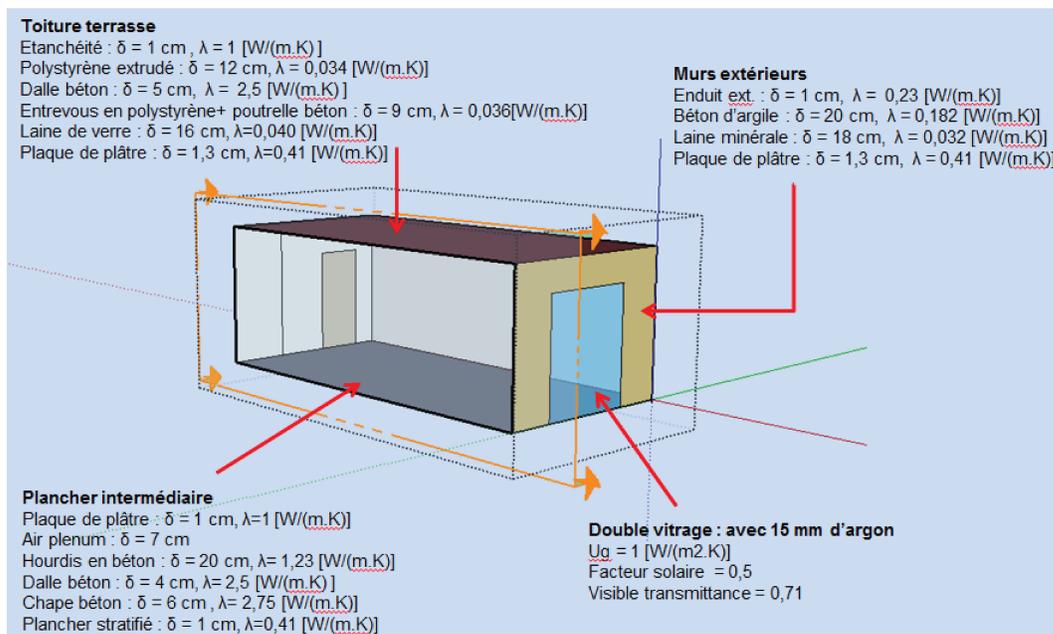


Figure C.6 – Maison POS-A - Positionnement des matériaux sur le modèle 3D.

2.2 Performances énergétiques attendues

L'évaluation des performances de cette maison a fait l'objet d'un autre livrable pour le projet COMEPOS (Stefanoiu et al. [99]). Nous en reprenons ici les conclusions. Le tableau C.3 ci-dessous présente le bilan énergétique prévisionnel de cette maison. On constate qu'elle est largement à énergie positive avec un bilan de $+35kWh_{EP}/m^2/an$.

On peut remarquer que le bois sera également utilisé en tant qu'énergie de chauffage. Cela entraîne un biais important en terme de mesures in-situ car il est particulièrement difficile de connaître la quantité d'énergie engendrée par la combustion du bois. Celle-ci dépend en effet du type de bois utilisé, de son humidité, etc. Ainsi, lorsque nous effectuerons des analyses sur la consommation énergétique, nous prendrons le parti de nous intéresser seulement à la

consommation électrique. Nous prendrons donc une énergie de chauffage équivalente à 75% de la consommation prévue, soit à $23,7 \text{ kWh}_{EP}/\text{m}^2/\text{an}$.

<i>Consommation [kWh_{EP}/m²]</i>	
Chauffage (75% Elec + 25% Bois)	31,6
ECS	11
Eclairage et électrodomestique	37
Auxiliaires	6,8
TOTAL CONSOMMATION	86,4
<i>Production [kWh_{EP}/m²]</i>	
Photovoltaïque	81,3
Eolien	40
TOTAL PRODUCTION	121,5
BILAN ENERGETIQUE	35

Table C.3 – Bilan énergétique prévisionnel

Annexe D

Requêtes http pour les webserveurs

Table des matières

1	Accès à l'aide sur les différentes requêtes disponibles	211
2	Accès à la liste des bâtiments dont l'historique est disponible	212
3	Accès aux noms des services et des variables disponibles	212
E	Facteurs traités lors de l'analyse de Morris	215

Ici des exemples de requêtes fonctionnant pour la récupération des données de la maison POS-A sont présentés. Cependant, afin d'avoir accès aux données, et dans un soucis de confidentialité, il est nécessaire d'obtenir un nom d'utilisateur ("NomUser") et un mot de passe ("PasswordUser").

1 Accès à l'aide sur les différentes requêtes disponibles

<http://vestaenergy.vesta-system.fr/COMEPOS-WS/service/getHelp.php?user=NomUser&password>PasswordUser>

2 Accès à la liste des bâtiments dont l'historique est disponible

`http://vestaenergy.vesta-system.fr/COMEPOS-WS/service/getBuildingList.php?user=NomUser&password>PasswordUser`

3 Accès aux noms des services et des variables disponibles

`http://vestaenergy.vesta-system.fr/COMEPOS-WS/service/getAllServiceVariables.php?building=POSA&user=NomUser&password>PasswordUser&format=html`

Après la requête `getAllServiceVariables`, on obtient ceci un tableau tel que présenté en figure D.1.

Building	Floor	Zone	Service	Device name	Variable
POSA	R0	Chambre2	device_320	EtatVoletChambre2	SHUTTER_STATE
POSA	R0	Chambre2	device_92	TemperatureHumidite	indoor relative humidity
POSA	R0	Chambre2	device_92	TemperatureHumidite	indoor temperature
POSA	R0	Chambre3	device_45	CompteurTVChambre3	POWER
POSA	R0	Chambre3	device_109	ContactFenetre	opening detection
POSA	R0	Chambre3	device_324	EtatFenetreChambre3	CONTACT
POSA	R0	Chambre3	device_324	EtatFenetreChambre3	position
POSA	R0	Chambre3	device_323	EtatLampe16	ON_OFF
POSA	R0	Chambre3	device_321	EtatLampe17	ON_OFF
POSA	R0	Chambre3	device_322	EtatLampe38	ON_OFF
POSA	R0	Chambre3	device_180	EtatTVChambre3	ON_OFF
POSA	R0	Chambre3	device_325	EtatVoletChambre3	SHUTTER_STATE
POSA	R0	Chambre3	device_90	TemperatureHumidite	indoor relative humidity
POSA	R0	Chambre3	device_90	TemperatureHumidite	indoor temperature

Figure D.1 – Requête webservice : `getAllVariableServices`.

Accès à l'historique

- Au format automatique : `http://vestaenergy.vesta-system.fr/COMEPOS-WS/service/getVariableHistory.php?building=POSA&user=NomUser&password>PasswordUser&service=device_92&variable=indoorrelativehumidity&start=01/12/2015&end=31/12/2015`
- Au format CSV : `http://vestaenergy.vesta-system.fr/COMEPOS-WS/service/getVariableHistory.php?user=NomUser&password>PasswordUser&building=POSA&service=device_92&variable=indoorrelativehumidity&start=01/12/2015&end=31/12/2015&format=csv`

Par rapport à l'exemple ci-dessus, et pour que la requête `getVariableHistory` fonctionne, il faut rentrer `service=device_92` et `variable=indoor relative humidity`. Il est enfin important de noter que le nom des services et variables est sensible à la casse.

Annexe E

Facteurs traités lors de l'analyse de Morris

Table des matières

F Maison OSTWALD

221

<i>Facteur</i>	<i>Unité</i>
mineral wool thickness	m
mineral wool conductivity	W/m.K
air wall conductivity	W/m.K
PolystyreneXtrude thick- ness	m
ext coat conductivity	W/m.K
200mm clay concrete conductivity	W/m.K
glass wool40 conductivity	W/m.K
air wall thickness	m
solid slab 4cm thickness	m
glass wool32 conductivity	W/m.K
glass wool32 density	kg/m ³
interior coat conductivity	W/m.K
glass wool40 thickness	m

<i>Facteur</i>	<i>Unité</i>
screed 5cm thickness	m
solid slab 16cm density	kg/m ³
interior coat density	kg/m ³
solid slab 4cm density	kg/m ³
tiles conductivity	W/m.K
solid slab 5cm thickness	m
screed 6cm thickness	m
interior coat thickness	m
entrevous conductivity	W/m.K
entrevous thickness	m
ext coat density	kg/m ³
solid slab 16cm thickness	m
ext coat thickness	m
solid slab 4cm conductivity	W/m.K
screed 6cm density	kg/m ³
Hourdis density	kg/m ³
air wall density	kg/m ³
200mm clay concrete thickness	m
screed conductivity	W/m.K
PolystyreneXtrude conductivity	W/m.K
entrevous density	kg/m ³
solid slab 16cm conductivity	W/m.K
PolystyreneXtrude density	kg/m ³
screed 5cm conductivity	W/m.K
STYROFOAM density	kg/m ³
screed thickness	m
mineral wool density	kg/m ³
etancheite multicouche density	kg/m ³
screed 6cm conductivity	W/m.K
Hourdis thickness	m
etancheite multicouche thickness	m

<i>Facteur</i>	<i>Unité</i>
screed 5cm density	kg/m ³
Hourdis conductivity	W/m.K
STYROFOAM thickness	m
STYROFOAM conductivity	W/m.K
tiles thickness	m
etancheite multicouche conductivity	W/m.K
glass wool40 density	kg/m ³
solid slab 5cm density	kg/m ³
200mm clay concrete density	kg/m ³
screed density	kg/m ³
Polystyrene density	kg/m ³
Polystyrene thickness	m
tiles density	kg/m ³
Polystyrene conductivity	W/m.K
glass wool32 thickness	m
solid slab 5cm conductivity	W/m.K
ventilation nocturne rdc bureau air changes	1/hr
tuiles density	kg/m ³
ventilation nocturne etage salon air changes	1/hr
solid slab 10cm density	kg/m ³
ventilation nocturne rdc ch air changes	1/hr
building block density	kg/m ³
waterproof coat density	kg/m ³
ROOFMATE density	kg/m ³
poutrelle entrevous density	kg/m ³
zone Inf air changes	1/hr
graviers density	kg/m ³
escalier Ventilation flow rate	m ³ /s

<i>Facteur</i>	<i>Unité</i>
garage Inf air changes	1/hr
window UFactor	W/m2.K
window Visible Transmittance	-
rdc bureau Ventilation flow rate	m3/s
rdc ch Ventilation air changes	1/hr
etage ch4 Ventilation air changes	1/hr
Occupation Zone Jour	People/Area
Occupation Zone Nuit	People/Area
Elec Eclairage Watts/Area	W/m2
etage salon Ventilation air changes	1/hr
pont t g flow rate	m3/s
pont t zone flow rate	m3/s
window Solar Heat Gain Coefficient	-
screed specific heat	J/kg.K
BA 13mm gypsum board density	kg/m3
building block specific heat	J/kg.K
ROOFMATE specific heat	J/kg.K
mineral wool specific heat	J/kg.K
BA 13mm gypsum board specific heat	J/kg.K
200mm clay concrete specific heat	J/kg.K
graviers conductivity	W/m.K
tuiles conductivity	W/m.K
poutrelle entrevous conductivity	W/m.K
solid slab 10cm conductivity	W/m.K
building block conductivity	W/m.K

<i>Facteur</i>	<i>Unité</i>
waterproof coat conductivity	W/m.K
ROOFMATE conductivity	W/m.K
BA 13mm gypsum board conductivity	W/m.K
graviers thickness	m
tuiles thickness	m
poutrelle entrevous thickness	m
solid slab 10cm thickness	m
building block thickness	m
waterproof coat thickness	m
ROOFMATE thickness	m
waterproof coat specific heat	J/kg.K
STYROFOAM specific heat	J/kg.K
air wall specific heat	J/kg.K
solid slab 16cm specific heat	J/kg.K
etancheite multicouche specific heat	J/kg.K
Hourdis specific heat	J/kg.K
entrevous specific heat	J/kg.K
solid slab 5cm specific heat	J/kg.K
PolystyreneXtrude specific heat	J/kg.K
screed 6cm specific heat	J/kg.K
screed 5cm specific heat	J/kg.K
Polystyrene specific heat	J/kg.K
solid slab 4cm specific heat	J/kg.K
graviers specific heat	J/kg.K
tuiles specific heat	J/kg.K
poutrelle entrevous specific heat	J/kg.K
ext coat specific heat	J/kg.K
glass wool32 specific heat	J/kg.K

<i>Facteur</i>	<i>Unité</i>
glass wool40 specific heat	J/kg.K
interior coat specific heat	J/kg.K
tiles specific heat	J/kg.K
BA 13mm gypsum board thickness	m
solid slab 10cm specific heat	J/kg.K
ventilation nocturne etage ch4 air changes	1/hr

Annexe F

Maison OSTWALD



Table des matières

1	Description thermique	221
2	Liste des capteurs installés	224

1 Description thermique



Fiche descriptive - Maison HANAU - Exe

Données générales

Composition famille : 2 adultes + 2 enfants
Typologie de maison : 2 niveaux - vide sanitaire
SHON RT = 163 m²
Surface habitable = 137 m²
Localisation : Ostwald (67)
Altitude = 139 m
Perméabilité = 0.40m³/(h/m²) sous 4Pa
Date de livraison : octobre 2016



Façade Sud-Ouest, Sud-Est

Résultats RT2012

Coefficient Cep_{SANS production ENR} = 54.6 kWhEP/m²SHONRT.an
Production ENR = 118.2 kWhEP/m²SHONRT.an
Coefficient Cep_{AVEC production ENR} = - 63.7 kWhEP/m²SHONRT.an
Bbio = 63.1 points
Tic = 26.2 °C



Enveloppe

Bâti :

Murs extérieurs :

- 25 cm de bloc thermopierre, R=2.27 m².K/W, ρ=350kg/m³
- 12 cm ITI en polystyrène graphité, λ =0.032 W/m.K

Rampants :

- 10 + 16 cm laine de verre, λ =0.032 W/m.K

Combles non aménagés :

- 40 cm de laine de roche soufflée, λ =0.045 W/m.K

Plancher bas :

- 16 cm poutrelles + hourdis polystyrène, Up=0.23 W/m².K, R=4 m².K/W, + 4 cm dalle de compression
- 8 cm mousse polyuréthane, λ =0.024 W/m.K
- Chape
- Carrelage

Plancher intermédiaire :

- 12 cm poutrelles + hourdis béton + 4 cm dalle de compression
- 3 cm mousse polyuréthane, λ =0.023 W/m.K
- Chape
- Carrelage

Traitement ponts thermiques :

- Front de dalle → plaquette céramique

Murs intérieurs :

- 4.5 cm laine de verre, λ =0.04 W/m.K

Vitrages :

Fenêtres et portes fenêtres :

- Menuiserie PVC, double vitrage 4/16/4
volet roulant PVC, émissivité=1.1

Velux :

- Double vitrage 4/15/5 Argon, Uw =1.5 W/m².K, ΔR =0.15 m².K/W, FS =0.62, TL =0.78

Porte d'entrée :

- N/A

Porte accès garage :

- N/A

Systemes

Technologie RENSON Systeme E+ installée qui traite à la fois la ventilation, le chauffage et l'ECS.

Chauffage :

- Energie → gaz
- Emetteurs → plancher-chauffant RDC/étage et radiateur sèche-serviette SDB étage (750W)
- Générateur de chaleur → ① **Chaudière gaz** à condensation modulable, puissance =6.1kW à 24.8kW + préchauffage générateur ②

ECS :

- Energie → électricité
- Distribution → ballon eau chaude intégré isolé =285L, isolé avec de la mousse polyuréthane
- Générateur de chaleur → ② **PAC air extérieur, air vicié/eau** + appoint générateur ①, $COP_{ECS}=4$, $COP_{CHAUFFAGE}=3.9$ ($t_{ext}=2^{\circ}C$ et $t_{eau}=35/30^{\circ}C$)

Ventilation :

- Bouches d'entrée d'air : Pièces de vie non humides, type hygroréglable sur menuiserie façade
- Bouches d'extraction : Etage =Pièces de vie + SDB, RDC =Cuisine + SDB, type commandées
- Régulation : occupation =taux de CO2/COV/HR, hors occupation =débit minimum
- Ventilateur : puissance nominale =67 W

Production/Stockage d'énergie

Photovoltaïque :

- 8 kWc installé en toiture orientation Sud-Est
- 32 modules, 250Wc/module
- Onduleur SMA

Stockage :

- Pas d'installation de stockage
- Revente à ES Energies Strasbourg

Régulation/pilotage

Les habitants sont équipés d'une tablette qui leur permet d'agir sur les systèmes.

Régulation :

- Chauffage, régulation avec thermostat d'ambiance et sonde extérieure
- Ventilation, 4 niveaux, normal, réduit, inoccupé, ou boost

Pilotage :

- Volets roulants, 4 positions, fonctionnement à la demande ou pilotage automatique en fonction de la température intérieure, température extérieure, ensoleillement, horloge astronomique et programme horaire

Instrumentation – Gestion CEA

- Sonde de confort : Température, Humidité relative, CO2 dans toutes les pièces
- Compteur électrique : Systeme E+ (PAC/ventilateur), Général, Prises, Production, Intensité général
- Compteur calorifique : ECS chaudière (appoint), Chauffage RDC PAC, Chauffage chaudière RDC, Chauffage chaudière, Température circuit ECS-chaudière, Volume circuit ECS-chaudière, Température circuit plancher-PAC, Volume circuit plancher-PAC
- Compteur d'eau : Volume eau chaude puisée, Température eau chaude puisée
- Compteur gaz : Volume de gaz

2 Liste des capteurs installés



Identification capteur	Unité	Modèle capteur	Zone	Descriptif
TEM_FAC_NOR_I_001	°C	Sonde STE 2000 - Delta Dore	exterieur	température façade Nord
FLX_SOL_SUD_I_001	W/m²	Sonde SE 2100 - Delta Dore	exterieur	rayonnement solaire
PRE_ATM_EXT_I_001	hPa	Données site web	exterieur	EXT pression atmosphérique
DIR_VNT_EXT_I_001		Données site web	exterieur	EXT direction du vent
VIT_VNT_EXT_I_001	km/h	Données site web	exterieur	EXT vitesse du vent
HUM_REL_EXT_I_001	%	Données site web	exterieur	EXT humidité relative
TEM_AIR_EXT_I_001	°C	Données site web	exterieur	EXT température air
TEM_CUI_S_I_001	°C	E4000 NanoSense	cuisine	température cuisine
HUM_CUI_S_I_001	%	E4000 NanoSense	cuisine	humidité relative cuisine
CO2_CUI_S_I_001	ppm	E4000 NanoSense	cuisine	quantité CO2 cuisine
TEM_SAL_NO_I_001	°C	E4000 NanoSense	salon	température salon
HUM_SAL_NO_I_001	%	E4000 NanoSense	salon	humidité relative salon
CO2_SAL_NO_I_001	ppm	E4000 NanoSense	salon	quantité CO2 salon
TEM_SAM_SO_I_001	°C	E4000 NanoSense	salle à manger	température salle à manger
HUM_SAM_SO_I_001	%	E4000 NanoSense	salle à manger	humidité relative salle à manger
CO2_SAM_SO_I_001	ppm	E4000 NanoSense	salle à manger	quantité CO2 salle à manger
TEM_ENT_NE_I_001	°C	E4000 NanoSense	entrée	température entrée
HUM_ENT_NE_I_001	%	E4000 NanoSense	entrée	humidité relative entrée
CO2_ENT_NE_I_001	ppm	E4000 NanoSense	entrée	quantité CO2 entrée
TEM_GA_E_I_001	°C	E4000 NanoSense	garage	température garage
HUM_GA_E_I_001	%	E4000 NanoSense	garage	humidité relative garage
CO2_GA_E_I_001	ppm	E4000 NanoSense	garage	quantité CO2 garage
TEM_SE_N_I_001	°C	E4000 NanoSense	sdb bas	température salle de bains bas
HUM_SE_N_I_001	%	E4000 NanoSense	sdb bas	humidité relative salle de bains bas
CO2_SE_N_I_001	ppm	E4000 NanoSense	sdb bas	quantité CO2 salle de bains bas
TEM_CEL_SE_I_001	°C	E4000 NanoSense	cellier	température cellier
HUM_CEL_SE_I_001	%	E4000 NanoSense	cellier	humidité relative cellier
CO2_CEL_SE_I_001	ppm	E4000 NanoSense	cellier	quantité CO2 cellier
TEM_CH1_SE_I_001	°C	E4000 NanoSense	chambre 1	température chambre 1
HUM_CH1_SE_I_001	%	E4000 NanoSense	chambre 1	humidité relative chambre 1
CO2_CH1_SE_I_001	ppm	E4000 NanoSense	chambre 1	quantité CO2 chambre 1
TEM_CH2_SO_I_001	°C	E4000 NanoSense	chambre 2	température chambre 2
HUM_CH2_SO_I_001	%	E4000 NanoSense	chambre 2	humidité relative chambre 2
CO2_CH2_SO_I_001	ppm	E4000 NanoSense	chambre 2	quantité CO2 chambre 2
TEM_CH3_NO_I_001	°C	E4000 NanoSense	chambre 3	température chambre 3
HUM_CH3_NO_I_001	%	E4000 NanoSense	chambre 3	humidité relative chambre 3
CO2_CH3_NO_I_001	ppm	E4000 NanoSense	chambre 3	quantité CO2 chambre 3
TEM_PAL_N_I_001	°C	E4000 NanoSense	palier haut	température palier haut
HUM_PAL_N_I_001	%	E4000 NanoSense	palier haut	humidité relative palier haut
CO2_PAL_N_I_001	ppm	E4000 NanoSense	palier haut	quantité CO2 palier haut
TEM_SDB_S_I_001	°C	E4000 NanoSense	sdb haut	température salle de bains haut
HUM_SDB_S_I_001	%	E4000 NanoSense	sdb haut	humidité relative salle de bains haut
CO2_SDB_S_I_001	ppm	E4000 NanoSense	sdb haut	quantité CO2 salle de bains haut
TEM_BUR_NE_I_001	°C	E4000 NanoSense	bureau	température bureau
HUM_BUR_NE_I_001	%	E4000 NanoSense	bureau	humidité relative bureau
CO2_BUR_NE_I_001	ppm	E4000 NanoSense	bureau	quantité CO2 bureau
ENE_ELE_GEN_I_001	kWh	TYWATT 5400 Delta Dore	maison globale	consommation électrique total
INT_ELE_GEN_I_001	A	TYWATT 5400 Delta Dore	maison globale	intensité électrique totale
ENE_ELE_CHA_I_001	kWh	TYWATT 5400 Delta Dore	système	consommation électrique de la PAC pour le chauffage
ENE_ELE_BTD_I_001	kWh	TYWATT 5400 Delta Dore	système	consommation électrique de la PAC pour l'ECS
ENE_ELE_USA_I_001	kWh	TYWATT 5400 Delta Dore	maison globale	consommation électrique des prises
ENE_ELE_PV_I_001	W	Onduleur Sunny Boy SMA	production	production des panneaux photovoltaïques
ENE_CHA_GEN_I_001	kWh	DN 25 + TYWATT 5200 Delta Dore	système	énergie fournie par la chaudière pour le chauffage
TEM_ECS_APP_I_001	°C	DN 25 + TYWATT 5200 Delta Dore	système	température circuit ECS-chaudière
VOL_ECS_APP_I_001	L	DN 25 + TYWATT 5200 Delta Dore	système	volume d'eau circuit ECS-chaudière
ENE_CHA_PAC_I_001	kWh	DN 25 + TYWATT 5200 Delta Dore	système	énergie fournie par la PAC pour le chauffage du RDC
TEM_CHA_PAC_I_001	°C	DN 25 + TYWATT 5200 Delta Dore	système	température circuit plancher-PAC
VOL_CHA_PAC_I_001	L	DN 25 + TYWATT 5200 Delta Dore	système	volume d'eau circuit plancher-PAC
ENE_CHA_RDC_I_001	kWh	DN 20 + TYWATT 5200 Delta Dore	système	énergie fournie par la chaudière et la PAC pour le chauffage du RDC
ENE_ECS_APP_I_001	kWh	DN 20 + TYWATT 5200 Delta Dore	système	énergie fournie par la chaudière pour l'ECS
ENE_ECS_PUI_I_001	kWh	DN 20 + TYWATT 5200 Delta Dore	système	énergie fournie par la chaudière et la PAC pour l'ECS
TEM_ECS_PUI_I_001	°C	DN 20 + TYWATT 5200 Delta Dore	système	température de l'eau puisée à la sortie du ballon
VOL_ECS_PUI_I_001	L	DN 20 + TYWATT 5200 Delta Dore	système	volume d'eau chaude puisée
VOL_GAZ_CHA_I_001	dm3	TYWATT 5100 Delta Dore	système	volume de gaz consommé par la chaudière

Résumé



Ces travaux de thèse portent sur le développement d'une nouvelle méthodologie pour l'identification de défauts de bâtiments performants et connectés afin d'aider à la garantie de performances. Nous avons dans un premier temps resitué nos travaux dans le contexte énergétique actuel en montrant le rôle majeur des bâtiments dans la réduction des consommations énergétiques. Nous avons ensuite présenté notre méthodologie en argumentant sur les techniques à utiliser avant d'effectuer un choix final. Cette méthodologie se compose de deux blocs principaux : le premier vise à réduire les incertitudes liées à l'occupant et à l'environnement et le second étudie l'écart entre la simulation et la mesure par une analyse de sensibilité couplée à un algorithme bayésien. Nous l'avons ensuite implémentée dans un outil que nous avons nommé REFATEC. Nous avons alors soumis notre méthodologie à différents tests dans des conditions idéales afin d'éprouver sa précision et son temps d'exécution. Cette étape a montré que la méthodologie est efficace mais montre quelques faiblesses dans le cas d'une saison estivale ou d'un défaut très localisé. Enfin, nous l'avons mise en situation face à un cas réel afin de traiter les nombreuses questions que soulèvent l'utilisation de mesures in-situ dans la perspective de la garantie de performances et de la détection de défauts, avec notamment la fiabilité des mesures et les incertitudes encore nombreuses qui doivent être traitées.

Mots clés : *Identification de défauts, bâtiment performant et connecté, analyse de sensibilité, algorithme bayésien, garantie de performances.*

Abstract



This thesis deals with the development of a new methodology for fault detection within smart high-performance buildings helping the performance guarantee. We first have placed our work in the current energy context by focusing on the major role of buildings in the decrease of energy consumption. Then we introduced our methodology and we argued about various techniques that could be used before making a choice. This methodology is made up of two main parts : the former reduces the uncertainties due to the occupant and the environment and the latter studies the gap between simulation and measurements thanks to a sensitivity analysis coupled with a bayesian algorithm. Then we implemented it within a tool that we named REFATEC. We carried out various tests in controlled conditions in order to evaluate its precision and its calculation time. This step showed that our methodology is effective but it has some difficulties when the studied period is during summer or when the faults are very located. is a very located fault. Eventually we confronted our methodology to a real case where we faced numerous questions that appear when dealing with measurements, especially their reliability and the uncertainties that still need to be taken care of, in the perspective of performance guarantee and fault detection.

Key words : *Fault detection, smart high-performance building, sensitivity analysis, bayesian algorithm, performance guarantee.*