

Contrôle intelligent de la domotique à partir d'informations temporelles multisources imprécises et incertaines

Pedro Chahuara

▶ **To cite this version:**

Pedro Chahuara. Contrôle intelligent de la domotique à partir d'informations temporelles multisources imprécises et incertaines. Intelligence artificielle [cs.AI]. Université de Grenoble, 2013. Français. NNT : . tel-00956372

HAL Id: tel-00956372

https://theses.hal.science/tel-00956372

Submitted on 6 Mar 2014

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

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

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Pedro CHAHUARA QUISPE

Thèse dirigée par **Michel VACHER**

préparée au sein **Laboratoire d'Informatique de Grenoble (LIG)**
et de **Ecole Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Contrôle intelligent de la domotique à partir d'informations temporelles multisources imprécises et incertaines

Thèse soutenue publiquement le **27 mars 2013**,
devant le jury composé de :

M. Patrick REIGNIER
Professeur des Universités, Grenoble INP, LIG, Président
M. François CHARPILLET
Directeur de Recherche INRIA, LORIA,Nancy, Rapporteur
M. Arnaud MARTIN
Professeur des Universités, Université de Rennes 1, Rapporteur
M. Alain ANFOSSO
Ingénieur, CSTB, Nice, Examinateur
Mme Christel VRAIN
Professeur des Universités, Université d'Orléans, Examinatrice
M. Michel VACHER
Ingénieur de Recherche CNRS HDR, LIG, Directeur de thèse
M. François PORTET
Maître de Conférences, Grenoble INP, LIG, Co-Encadrant de thèse



Remerciements

La thèse de doctorat, comme la plupart des activités de recherche, ne peut pas être seulement vue comme une tâche individuelle, l'aspect collaboration avec une équipe est fondamental. Lors de ces trois dernières années d'études qui ont représenté pour moi une des étapes les plus passionnantes de ma vie, j'ai eu l'opportunité d'être entouré de personnes très talentueuses dont les conseils m'ont beaucoup apporté, me permettant non seulement de conclure mes études de doctorat, mais aussi de donner un cap à ma carrière scientifique.

En premier lieu, je tiens à remercier Michel Vacher et François Portet pour leur soutien et leur patience comme directeurs de thèse. Leur expérience et leur motivation ont rendu possible les contributions scientifiques qui constituent l'ossature de cette thèse. Je remercie également les autres membres de l'équipe GETALP : lorsque j'exposais l'état de mes recherches lors des réunions animés par Laurent Besacier, les retours qui m'étaient faits m'ont beaucoup aidé à avancer. Je voudrais aussi remercier Benjamin Lecouteux avec qui j'ai échangé de nombreuses idées. L'expérience d'Anthony Fleury m'a été précieuse et les discussions que nous avons eues ont toujours été fructueuses, je tiens à le remercier.

J'aimerais remercier Arnaud Martin et François Charpillet pour avoir accepté de rapporter ma thèse. Je remercie egalement Patrick Reignier qui a accepté de présider le jury, ainsi que Christel Vrain, et Alain Anfosso pour leur participation au jury comme examinateurs.

Je remercie Brigitte Meillon, Nicolas Bonnefond, et Mathieu Gallisot pour leur aide lors des expérimentations dans l'appartement Domus. Un grand merci à l'ensemble de doctorants de l'équipe Getalp : Fréderic, David, Sarah, Marwen, Quang et les doctorants de l'équipe MRIM : Andrew, Nadia, Mateusz, Johan, Path qui ont crée une bonne ambiance de travail et permis des moments de convivialité très agréables.

Je remercie finalement Christelle pour son soutien et sa compréhension lors de la finalisation de cette thèse.

Table des matières

Table des matières 8

Résumé 9

Abstract 11

Table des figures 13

Liste des tableaux 15

1 Introduction 17
- 1.1 Motivation . . . 17
- 1.2 Projet ANR Sweet-Home . . . 18
- 1.3 Définition du problème . . . 19
- 1.4 Objectifs du travail de thèse . . . 20
- 1.5 Plan du manuscrit . . . 21

2 État de l'art 23
- 2.1 Intelligence Ambiante . . . 23
  - 2.1.1 Environnements intelligents . . . 24
  - 2.1.2 Domotique . . . 26
  - 2.1.3 Maisons intelligentes . . . 27
  - 2.1.4 Projets de maisons intelligentes . . . 29
    - 2.1.4.1 House_n - Massachusetts Institute of Technology (MIT) . . . 30
    - 2.1.4.2 CASAS - Washinton State University . . . 31
    - 2.1.4.3 The Intelligent Dormitory (iSpace) - The University of Essex . . . 31
    - 2.1.4.4 GER'HOME - CSTB . . . 32
    - 2.1.4.5 Aging In Place - University of Missouri . . . 33
    - 2.1.4.6 CompanionAble - UE . . . 34
    - 2.1.4.7 SOPRANO - UE . . . 34
    - 2.1.4.8 Smart hoMes for All / SM4ALL - UE . . . 35
    - 2.1.4.9 HomeCare - LAAS Toulouse . . . 36
    - 2.1.4.10 Positionnement de nos travaux par rapport à ces projets . . . 37
- 2.2 Analyse de contexte . . . 38
  - 2.2.1 Définition du contexte dans le cadre de l'Intelligence ambiante . . . 38
  - 2.2.2 Sensibilité au contexte . . . 39
  - 2.2.3 Travaux sur la reconnaissance des situations en modélisation du contexte 40
  - 2.2.4 Éléments du contexte . . . 43
    - 2.2.4.1 Localisation de l'habitant . . . 43
    - 2.2.4.2 Reconnaissance de l'activité . . . 45

2.2.5 Reconnaissance de la parole et des sons comme support à la reconnaissance du contexte . . . 47
2.3 Prise de décision dans les environnements perceptifs . . . 49
2.3.1 Définitions . . . 49
2.3.2 Défis des systèmes de prise de décision . . . 50
2.3.2.1 Prise de décision en contexte . . . 51
2.3.3 Travaux existants sur la prise de décision . . . 51
2.3.3.1 Utilisation de systèmes fondés sur des règles logiques . . . 52
2.3.3.2 Utilisation de systèmes fondés sur des ontologies . . . 52
2.3.3.3 Utilisation de systèmes fondés sur des méthodes statistiques ou probabilistes . . . 53
2.3.3.4 Conséquences pour un choix optimal . . . 53
2.4 Synthèse et conséquences . . . 53

3 Contrôle intelligent de la domotique 57
3.1 Environnement domotique de référence . . . 58
3.1.1 Informations d'entrée . . . 59
3.1.2 Informations de sortie . . . 60
3.1.2.1 Commandes domotiques . . . 60
3.1.2.2 Message vocaux . . . 60
3.1.2.3 Messages à destination des systèmes spécialisés . . . 60
3.2 Scénarios . . . 61
3.3 Inférence et hiérarchie des informations . . . 62
3.4 Architecture fonctionnelle du contrôleur intelligent . . . 62
3.4.1 Structure générale . . . 63
3.4.1.0.1 Ontologie . . . 64
3.4.1.0.2 Acquisition de Données . . . 65
3.4.1.0.3 Base de Faits . . . 65
3.4.1.0.4 Détection de Localisation . . . 65
3.4.1.0.5 Reconnaissance d'activité . . . 65
3.4.1.0.6 Reconnaissance d'ordres . . . 65
3.4.1.0.7 Base de Règles . . . 66
3.4.1.0.8 Détection de Situations . . . 66
3.4.1.0.9 Prise de décision . . . 66
3.4.1.0.10 Gestion des actionneurs . . . 66
3.4.2 Architecture du logiciel . . . 66
3.4.3 Interaction entre modules . . . 69
3.4.4 Interaction avec les systèmes externes . . . 69
3.5 Représentation des connaissances . . . 69
3.5.1 Ontologie de bas niveau . . . 69
3.5.2 Ontologie de haut niveau . . . 70
3.5.3 Relation entre les deux ontologies . . . 70
3.6 Synthèse . . . 72

4 Inférence des informations 73
4.1 Des données brutes aux informations de haut niveau . . . 74
4.2 Méthodes d'inférence . . . 75
4.2.1 Approches statistiques . . . 76
4.2.1.1 Les Séparateurs à Vaste Marge (SVM) . . . 76
4.2.1.2 Les réseaux dynamiques et la propagation d'activation . . . 77
4.2.1.3 Modèle de Markov caché . . . 79

4.2.1.4 Modèle de Markov caché hiérarchique . . . 82
4.2.1.5 Champs aléatoires de Markov . . . 83
4.2.2 Approches logiques . . . 85
4.2.2.1 Concepts de la logique pour l'intelligence artificielle . . . 86
4.2.2.2 Systèmes d'inférence à base de règles logiques . . . 86
4.2.2.3 Programmation logique inductive (ILP) . . . 87
4.2.2.4 Logique floue . . . 89
4.2.2.5 Raisonnement par ontologies . . . 91
4.2.2.6 SWRL . . . 94
4.2.3 Approches statistique-relationnelles . . . 95
4.2.3.1 Modèle de Markov caché logique . . . 95
4.2.3.2 Réseaux logiques de Markov . . . 97
4.2.4 Discussion des méthodes d'inférence . . . 102
4.3 Localisation de l'habitant . . . 103
4.3.1 Formalisation . . . 103
4.3.2 Architecture du réseau dynamique . . . 104
4.3.2.1 Évolution du réseau dynamique . . . 104
4.3.2.2 Propagation de l'activation . . . 105
4.3.2.3 Calcul des relations entre les différentes couches du réseau . . . 106
4.4 Reconnaissance d'activités . . . 107
4.4.1 Reconnaissance d'activités par MLN . . . 107
4.4.1.1 Information inférée . . . 108
4.4.1.2 Calcul des vecteurs d'attributs . . . 108
4.4.2 Adaptation des réseaux logiques de Markov à la reconnaissance d'activités 109
4.5 Reconnaissance de situations . . . 111
4.5.1 Les situations dans le cadre de l'habitat intelligent . . . 111
4.5.2 Application de SWRL pour la reconnaissance de situations . . . 112
4.6 Conclusion . . . 113

5 Prise de décision 115
5.1 Définitions . . . 115
5.2 Méthodes classiques de prise de décision . . . 117
5.2.1 Programmation linéaire multicritère . . . 117
5.2.2 Systèmes experts . . . 119
5.2.3 Arbres de décision . . . 120
5.3 Diagrammes d'influence . . . 121
5.4 Modélisation d'un diagramme d'influence par MLN . . . 123
5.4.1 Prise en compte de l'incertitude de l'évidence . . . 125
5.5 Synthèse . . . 126

6 Expérimentations et résultats 127
6.1 Environnement perceptif utilisé pour les évaluations . . . 128
6.2 Première expérimentation : constitution d'un corpus multimodal . . . 129
6.2.1 Démarche expérimentale . . . 129
6.2.1.1 Protocole expérimental . . . 129
6.2.1.2 Contraintes imposées aux sujets . . . 130
6.2.1.3 Activités à réaliser . . . 130
6.2.2 Annotation du corpus recueilli . . . 131
6.2.3 Enregistrements . . . 132
6.3 Deuxième expérimentation : évaluation du contrôleur intelligent . . . 133
6.3.1 Démarche expérimentale . . . 133

6.3.2 Grammaire des ordres domotiques 134
6.3.3 Logiciel PATSH 134
6.4 Évaluation de la méthode de localisation 136
6.4.1 Calcul de poids dans le cas du HIS 136
6.4.2 Calcul de poids dans le cas de DOMUS 138
6.4.3 Expérimentations et résultats 138
6.4.4 Discussion sur les résultats de la localisation 139
6.5 Évaluation de la méthode de Reconnaissance d'Activité 140
6.5.1 Évaluation 1 140
6.5.2 Évaluation 2 141
6.5.3 Discussion sur la reconnaissance d'activité 143
6.6 Évaluation de la reconnaissance de situations et commandes vocales 143
6.7 Évaluation de la méthode de prise de décision 145
6.7.1 Modèle pour la prise de décision 145
6.7.2 Determination des informations contextuelles 147
6.7.3 Résultats avec le modèle de décision 148
6.7.4 Discussion sur les résultats de prise de décision 148

7 Conclusion et perspectives **151**

Bibliographie **155**

Bibliographie personelle **169**

Index **171**

Annexes **173**

A Scénario de l'expérimentation 1 : constitution d'un corpus multimodal **173**

B Scénario de l'expérimentation 2 : évaluation du contrôleur intelligent **179**

Résumé

La Maison Intelligente est une résidence équipée de technologie informatique qui assiste ses habitants dans les situations diverses de la vie domestique en essayant de gérer de manière optimale leur confort et leur sécurité par action sur la maison. La détection des situations anormales est un des points essentiels d'un système de surveillance à domicile. Ces situations peuvent être détectées en analysant les primitives générées par les étages de traitement audio et par les capteurs de l'appartement. Par exemple, la détection de cris et de bruits sourds (chute d'un objet lourd) dans un intervalle de temps réduit permet d'inférer l'occurrence d'une chute.

Le but des travaux de cette thèse est la réalisation d'un contrôleur intelligent relié à tous les périphériques de la maison et capable de réagir aux demandes de l'habitant (par commande vocale) et de reconnaître des situations à risque ou de détresse. Pour accomplir cet objectif, il est nécessaire de représenter formellement et raisonner sur des informations, le plus souvent temporelles, à des niveaux d'abstraction différents. Le principal défi est le traitement de l'incertitude, de l'imprécision, et de l'incomplétude, qui caractérisent les informations dans ce domaine d'application. Par ailleurs, les décisions prises par le contrôleur doivent tenir compte du contexte dans lequel un ordre est donné, ce qui nous place dans l'*informatique sensible au contexte*. Le contexte est composé des informations de haut niveau telles que la localisation, l'activité en cours de réalisation, la période de la journée.

Les recherches présentées dans ce manuscrit peuvent être divisées principalement en trois axes : la réalisation des méthodes d'inférence pour acquérir les informations du contexte (notamment, la localisation de l'habitant et l'activité en cours) à partir des informations incertaines, la représentation des connaissances sur l'environnement et les situations à risque, et finalement la prise de décision à partir des informations contextuelles. La dernière partie du manuscrit expose les résultats de la validation des méthodes proposées par des évaluations menées sur la plateforme expérimentale Domus.

Mots clés : Intelligence ambiante, environnements perceptifs, représentation sémantique, prise de décision, reconnaissance d'activités, ontologies, réseau logique de Markov.

Abstract

A smart home is a residence featuring ambient intelligence technologies in order to help its dwellers in different situations of common life by trying to manage their comfort and security through the execution of actions over the effectors of the house. Detection of abnormal situations is paramount in the development of surveillance systems. These situations can be detected by the analysis of the traces resulting from audio processing and the data provided by the network of sensors installed in the smart home. For instance, detection of cries along with thuds (fall of a heavy object) in a short time interval can help to infer that the resident fell.

The goal of the research presented in this thesis is the implementation of an intelligence controller connected with the devices in the house and able to react to user's commands (through vocal interfaces) and recognize dangerous situations. In order to fulfill this goal, it is necessary to create formal representation and to develop reasoning mechanism over information that is often temporal and having different levels of abstraction. The main challenge is the processing of the uncertainty, the imprecision, and the incompleteness that characterise this domain of application. Moreover, the decisions taken by the intelligent controller must consider the context in which a user command is given, so this work is made in the area of *Context Aware Computing*. Context includes high level information such as the location of the dweller, the activity she is making, and the time of the day.

The research works presented in this thesis can be divided mainly in three parts : the implementation of inference methods to obtain context information (namely, location and activity) from uncertain information, knowledge representation about the environment and dangerous situations, and finally the development of decision making models that use the inferred context information. The last part of this thesis shows the results from the validation of the proposed methods through experiments performed in an experimental platform, the Domus apartment.

Key Words : Ambient intelligence, smart environments, semantic representation, decision making, activity recognition, ontologies, Markov logic networks.

Table des figures

2.1 Relation entre l'intelligence ambiante et l'intelligence artificielle selon Ramos et coll. (2008) . . . 25
2.2 Exemple de système dynamique d'une maison intelligente . . . 28
2.3 Environnement intelligent du projet CASAS (Cook et Das, 2007) . . . 32
2.4 Architecture de l'intergiciel OpenAAL . . . 35
2.5 Prise de décision en contexte . . . 51

3.1 Interactions dans le système domotique . . . 58
3.2 Informations intermédiaires inférées à partir informations d'entrée (PIR=détecteur infra-rouge, CP=contact de porte) . . . 63
3.3 Modules du contrôleur intelligent . . . 64
3.4 Diagramme de classes du contrôleur intelligent . . . 68
3.5 Ontologies représentant l'environnement perceptif . . . 71

4.1 Réseau dynamique proposé par Niessen pour diminuer l'ambiguïté sonore . . . 79
4.2 Exemple de modèle de Markov caché . . . 80
4.3 Modèle de Markov Caché pour l'activité « préparer un repas » . . . 81
4.4 Exemple de modèle de Markov caché hiérarchique à quatre niveaux . . . 83
4.5 Modèle de Markov Caché Hiérarchique pour l'activité « préparer un repas » . . . 84
4.6 Fonctions d'appartenance selon la température d'un habitat . . . 89
4.7 Un système d'inférence floue général (Mendel, 1995) . . . 90
4.8 Obtention d'un réseau de Markov à partir d'un MLN . . . 97
4.9 Réseau de Markov créé lors de l'inférence d'un MLN . . . 98
4.10 Exemple de réseau dynamique . . . 105
4.11 Fenêtrage temporel utilisé pour l'obtention des attributs à partir des données fournies par les capteurs. . . . 109

5.1 Arbre de décision pour la reconnaissance d'un ordre vocal. . . . 121
5.2 Diagramme d'influence pour une décision après la reconnaissance d'un ordre vocal. . . . 123

6.1 Plan de l'appartement DOMUS et disposition des capteurs. . . . 128
6.2 Exemple d'annotation du corpus avec Advene . . . 132
6.3 Modules fonctionnels du système PATSH. . . . 135
6.4 Plan de l'appartement et position des capteurs. . . . 136

Liste des tableaux

3.1 Informations en entrée 59
3.2 Sorties vers les actionneurs 60
3.3 Messages vocaux à destination de l'habitant 61

4.1 Syntaxe et sémantique de la logique de description 92
4.2 Attributs utilisés pour la classification 110

6.1 Types d'annotation 131
6.2 Grammaire des ordres domotiques 134
6.3 Estimation de $P(Loc|Mic)$ dans le cas de HIS 137
6.4 Exactitude(%) avec plusieurs combinaisons de sources 138
6.5 Taux de bonnes classifications avec SVM et MLN (avec et sans classe rejet) . . . 141
6.6 Exactitude général, précision et rappel pour chaque classe 142
6.7 Matrice de confusion pour la reconnaissance d'activité lors de la prise de décision 147
6.8 Pourcentage d'instances de décision avec l'action correcte 148

CHAPITRE 1

Introduction

1.1 Motivation

La maison est un lieu particulièrement important pour tout un chacun, étymologiquement il s'agit du lieu où l'on reste, où l'on revient, du lieu de sédentarisation. La majorité des individus, et plus particulièrement les personnes âgées, passent beaucoup de leur temps à domicile, d'où l'influence considérable de l'habitat sur la qualité de vie. L'amélioration du sentiment de sécurité et de confort dans l'habitat apparaît donc comme une tâche d'une grande importance sociale. Ces dernières années, l'informatique a été appliquée à la création de *maisons intelligentes* dans le but d'améliorer les conditions de vie des gens lorsqu'ils sont à leur domicile. Une maison intelligente est définie comme une résidence équipée de technologies d'informatique ambiante qui vise à assister l'habitant dans les situations diverses de la vie domestique.

Les maisons intelligentes ont la capacité d'augmenter le confort de l'habitant à travers, par exemple, des interfaces naturelles pour piloter la lumière, la température ou les différents appareils électroniques. La gestion des ressources énergétiques est un autre enjeu des maisons intelligentes. Il est ainsi possible de mettre en veille les dispositifs de chauffage quand les habitants sont absents ou adapter automatiquement l'utilisation des ressources électriques en fonction des besoins des résidents afin de diminuer les gaspillages de ressources énergétiques. En outre, un autre but essentiel de l'application des technologies d'information aux maisons est la protection des individus. Cela est rendu possible par des systèmes capables d'anticiper des situations potentiellement dangereuses ou de réagir aux événements mettant en danger l'intégrité des personnes.

Les bénéficiaires de ces innovations peuvent être des individus autonomes mais également des personnes fragiles ayant une capacité limitée de mouvement. Par exemple, les personnes âgées ayant une autonomie limitée pourraient profiter des applications des maisons intelligentes pour faciliter leur vie quotidienne ou rester en contact avec leurs proches. Actuellement, les changements démographiques provoqués par le vieillissement de la population et l'augmentation du nombre de personnes âgées vivant seules ont un impact social et économique important au sein de la société. À cet égard, l'usage de la technologie représente une grande opportunité pour les personnes âgées vivant seules. Les systèmes intelligents peuvent rappeler aux habitants lorsqu'ils doivent prendre leurs médicaments, faciliter leur mise en communication avec l'extérieur ou même alerter les proches ou les services d'urgence si

la personne tombe par accident.

Les travaux de recherche concernant les maisons intelligentes ont fait l'objet d'un fort engouement au cours des dernières décennies à cause de l'enjeu sociétal correspondant tourné vers l'amélioration de la qualité de vie. Pourtant, les nombreux projets menés dans l'ensemble des pays n'ont pas suffi à relever les nombreux défis posés. Il est par ailleurs nécessaire de mettre en œuvre des interfaces utilisateur faciles d'utilisation pour des personnes peu ou pas habituées à interagir avec des technologies d'information conventionnelles. Les systèmes à développer doivent donc apparaître non pas comme de nouvelles contraintes perturbant le déroulement des activités quotidiennes mais bien comme de réels outils d'assistance qui facilitent la vie des personnes.

De même, certains problèmes scientifiques et techniques non encore résolus rendent difficile la mis en place des technologies de maisons intelligentes en situation réelle. Ces difficultés sont liées à l'implémentation des systèmes correspondants et aux méthodes informatiques sur lesquelles ils s'appuient. Étant donnée l'importance que va prendre la création de technologies domotiques et d'intelligence ambiante dans les prochaines années, nous estimons pertinente l'étude des techniques et méthodes pouvant rendre possible l'implémentation des maisons intelligentes.

1.2 Projet ANR Sweet-Home

Les recherches de cette thèse ont été menées dans le cadre du projet SWEET-HOME (Vacher et coll., 2011a). Ce projet est financé grâce à l'appel VERSO de l'Agence Nationale pour la Recherche de 2009. Il a débuté en novembre 2009 et se terminera en mai 2013. Ce projet fait suite à différents projets conduits par le laboratoire depuis une dizaine d'années :

- les projets RESIDE-HIS (IMAG 2000-2001) et DESDHIS (ACI Santé du Ministère de la Recherche 2002-2004) consacré à la détermination d'une éventuelle situation de détresse d'une personne à domicile par analyse des sons et de la parole enregistrés dans chacune des pièces de l'appartement ;
- le projet DESDHIS2 (collaboration avec le laboratoire TIMC-IMAG 2005-2008) consacré à la reconnaissance des appels de détresse dans l'appartement et à la surveillance des Activités de la Vie Quotidienne en prenant en compte les données issues des microphones et des capteurs placés dans l'appartement de test ainsi que d'un capteur de mouvement porté par la personne.

Le projet SWEET-HOME a pour but la conception d'un système de contrôle de la maison intelligente incorporant une commande vocale. Son propos est centré sur trois aspects :

1. fournir une assistance par une interaction homme-machine naturelle (voix et commande vocale),
2. faciliter l'inclusion sociale,
3. et assurer la sécurité de l'habitant par la détection de situations de détresse.

Si ces objectifs sont atteints, l'habitant pourra piloter son environnement depuis n'importe quel endroit de son logement, à tout moment et de la façon la plus naturelle.

Les utilisateurs cibles sont des personnes âgées mais qui sont encore pleinement autonomes. Deux raisons expliquent ce choix. Tout d'abord, un système domotique sera plus intéressant financièrement s'il est utilisé sur une grande période et pas uniquement lorsqu'une perte d'autonomie apparaît. Deuxièmement, en cas de perte d'autonomie, la personne continuera à utiliser son propre système moyennant quelques adaptations requises par cette nouvelle situation, par exemple l'utilisation d'un fauteuil roulant, sans avoir à faire face simultanément à sa perte d'autonomie et aux changements apportés dans sa vie par l'appartement intelligent.

Pour mesurer le niveau d'acceptation de cette technologie, une évaluation qualitative préliminaire a été menée sur un panel composé de 8 personnes d'âge compris entre 71 et 88 ans, accompagnées chacune d'un proche (membre de leur famille ou ami), et de 3 professionnels de santé (Portet et coll., 2013). Ces personnes ont participé à une enquête en co-découverte dans un appartement intelligent complètement équipé durant laquelle ont alterné des phases d'entretien et de manipulation avec un magicien d'Oz. Les quatre points importants du projet ont été abordés : la commande vocale, la communication avec le monde extérieur, l'interruption par le système lors d'une activité en cours et l'utilisation d'un agenda électronique. Dans chaque cas, l'utilisation d'une commande vocale a été préférée à des solutions jugées plus intrusives comme la caméra vidéo. Cette étude a donc confirmé les résultats d'autres études (López-Cózar et Callejas, 2010) qui ont montré l'intérêt des technologies audio pour améliorer la vie des personnes dépendantes et fragiles. Bien entendu, le respect de la vie privée impose une analyse au fil de l'eau de l'information et exclut tout enregistrement du signal audio. De plus, le système de reconnaissance de la parole ne permet de reconnaître qu'un jeu restreint de phrases nécessaires pour le pilotage du système ce qui limite la reconnaissance de phrase à caractère personnel.

1.3 Définition du problème

La réalisation des maisons intelligentes a été au cœur de plusieurs projets de recherche lors de ces dernières années. D'énormes progrès ont été accomplis grâce aux avancées en intelligence artificielle, à la miniaturisation de dispositifs électroniques pour la domotique et au développement des réseaux de communication, malgré tout, il reste encore plusieurs défis à surmonter pour rendre possible l'implémentation des maisons intelligentes en situation réelle.

Les progrès dans l'analyse automatique de vidéos sont suffisants pour permettre leur application dans les espaces intelligents, mais leur utilisation dans la vie courante au domicile est considérée trop intrusive comme l'ont démontré différentes études. De la même manière, les capteurs embarqués sont de plus en plus utilisés pour la surveillance médicale à domicile – surtout pour les personnes à autonomie réduite – parce qu'ils offrent des informations directes concernant la position de la personne et ses signes vitaux, toutes in-

formations très utiles pour identifier l'état dans lequel se trouve la personne. Cependant, ces capteurs peuvent être rejetés parce qu'ils représentent une contrainte trop forte pour les utilisateurs, celle du port permanent de l'objet, ce qui exclut un usage général. Les technologies intrusives sont généralement plus acceptées lorsque la personne y trouve une réelle compensation d'un handicap. Or notre projet cible des personnes âgées mais valides. Nous avons donc choisi l' utilisation des capteurs domotiques standards (capteurs de présence, contacts de porte, microphones, etc.) qui sont des sources d'information indirecte permettant d'évaluer la situation courante de l'habitant. Mais l'emploi de ces capteurs produit davantage d'incertitudes et d'imprécisions lors de l'inférence d'une information. Par exemple, dans le cas de la reconnaissance de la situation de l'habitant, connaître la position de la personne et sa localisation précise dans une pièce permettrait de reconnaître avec exactitude l'activité en cours ; tandis que les informations des sources indirectes indiquent comme localisation une pièce sans spécifier un point spécifique – c'est à dire, avec **imprécision** – et nécessitent d'être complémentées par l'utilisation d'objets et informations sonores pour déduire une situation avec un certain degrée d'**incertitude**.

C'est à partir d'informations de cette nature que le système domotique doit évaluer la nécessité d'exécuter une commande et ensuite décider de l'action qui sera la plus pertinente pour garantir le bien-être et la sécurité de l'habitant. Outre les imperfections des informations d'entrée, il existe également une incertitude concernant les relations entre les éléments du processus de décision. Ainsi, même si une situation peut être reconnue avec certitude, il peut encore subsister une incertitude quant à l'action qui correspondra le mieux à cette situation. Par ailleurs, les situations à reconnaître nécessitent l'implémentation de plusieurs méthodes, chacune adaptée à une tâche spécifique du raisonnement.

1.4 Objectifs du travail de thèse

Nos recherches portent sur la réalisation d'un contrôleur intelligent pour piloter la domotique à partir des données reçues d'un réseau domotique et d'un système d'analyse sonore ayant la capacité de reconnaître les ordres vocaux. Le contrôleur est l'élément principal du système domotique et constitue le cœur du projet SWEET-HOME, puisque il a le rôle essentiel de mettre en œuvre les mécanismes nécessaires pour agir face aux besoins de l'utilisateur. Dans nos recherches, nous supposons qu'il y a seulement une personne qui interagira avec le système dans l'environnement. Les actions exécutées par le contrôleur se produisent en réponse aux commandes vocales prononcées par l'habitant ou de manière proactive pour assurer sa sécurité et son confort.

La réalisation du contrôleur nécessite l'implémentation de méthodes de raisonnement et d'inférence de connaissances. Par conséquence, c'est à travers la création de ces composants que l'intelligence ambiante est mise en place dans le système domotique issu du projet SWEET-HOME. Les objectifs de nos recherches peuvent être divisés essentiellement selon les axes suivants :

– Proposer un modèle de représentation des connaissances pour un environnement in-

telligent qui puisse être facile à comprendre pour les différentes personnes impliquées dans le développement d'une maison intelligente. Le modèle à développer doit être capable également d'exprimer, facilement et sans ambiguïté, les relations complexes intervenant dans un environnement perceptif. Les connaissances devant être partagées par plusieurs modules d'inférence, il est donc nécessaire d'adopter un modèle qui offre un vocabulaire commun et qui soit basé sur un langage standard.

– Proposer des méthodes de l'intelligence artificielle pour inférer des informations de haut niveau à partir des données captées sur le réseau domotique. Ces informations sont spécifiquement la localisation et l'activité en cours. Nous considérons que ces deux éléments sont les plus importants pour mieux caractériser la situation courante dans un environnement perceptif.
– Définir un modèle pour décrire des situations d'intérêt dans l'environnement. Ces situations peuvent être des situations d'urgence représentant un risque pour l'habitant, ou simplement des situations au cours desquelles l'exécution d'une commande domotique peut améliorer le confort de l'habitant.
– Proposer une méthode de prise de décision pour l'exécution des commandes domotiques qui considère l'incertitude propre aux informations utilisées dans les environnements perceptifs. De plus, cette méthode doit mettre clairement en évidence le processus de décision et les éléments caractéristiques de la théorie de la décision tels que le risque, l'activité et le coût des actions.

1.5 Plan du manuscrit

Les définitions les plus importantes en rapport avec la maison intelligente sont données dans le chapitre 2. Dans ce chapitre, nous commencerons par aborder les relations existantes entre l'intelligence ambiante et la domotique avant de décrire les systèmes d'environnement intelligent en approfondissant plus particulièrement le cas de la maison intelligente. Ensuite nous décrirons les projets de maisons intelligentes de l'état de l'art qui nous semblent les plus pertinents et en rapport avec nos recherches. Toujours dans le même chapitre, après un certain nombre de définitions, nous ferons une revue de l'état de l'art en ce qui concerne le contexte et la prise de décision. Nous aborderons dans ce cadre les méthodes de localisation de l'habitant dans un environnement perceptif, la reconnaissance d'activités de vie quotidienne et la reconnaissance de situations.

Le chapitre 3 présente l'architecture du système domotique défini dans le cadre du projet SWEET-HOME et décrit la place du contrôleur intelligent dans le système. Les détails concernant l'interaction du contrôleur avec les autres composants du système sont donnés dans ce chapitre. Les scénarios types que nous présentons permettent d'appréhender le fonctionnement du système envisagé. Enfin, l'architecture fonctionnelle et logicielle est présentée ainsi que la représentation des connaissances par une ontologie à deux niveaux.

Le chapitre 4 aborde les méthodes d'inférence que nous avons implémentées pour atteindre les trois premiers objectifs de cette thèse. Cette partie a pour but de détailler les

méthodes de localisation de l'habitant, la reconnaissance d'activité, et la reconnaissance de situations. Avant de décrire nos implémentations, et afin de justifier les choix que nous avons faits, nous faisons le tour des méthodes susceptibles d'être employées pour ces tâches de raisonnement. Nous présentons ensuite les résultats que nous avons obtenus avec l'application de ces méthodes dans des environnements réels.

Le chapitre 5 est consacré à la prise de décision. La démarche est la même que celle du chapitre précédent, nous présentons les méthodes disponibles dans l'état de l'art et ensuite nous exposons notre implémentation. De la même manière, à la fin de ce chapitre nous présentons les résultats des expérimentations que nous avons menées sur la prise de décision grâce à des expérimentations dans un environnement réel.

Le chapitre 6 détaille les expérimentations faites pour valider le contrôleur intelligent en situations réalistes dans l'appartement DOMUS.

Pour finir nous présentons les conclusions de ce travail de thèse et nos perspectives de recherche.

CHAPITRE 2

État de l'art

Le début de ce chapitre est consacré à la description de l'intelligence ambiante qui est le sous-domaine de l'informatique dans lequel se situe notre recherche. Nous aborderons en particulier la relation entre l'intelligence ambiante et les autres disciplines, les concepts les plus importants qui s'y rattachent avant de décrire celui de la maison intelligente qui a servi de cadre à nos travaux. Ensuite, un rapide tour d'horizon permettra d'aborder les différents projets de maison intelligente existants et de les situer par rapport à nos travaux. La dernière partie du chapitre dressera un état de l'art de la reconnaissance de situations avant de présenter en détail la prise de décision en contexte.

2.1 Intelligence Ambiante

Du fait des évolutions technologiques et de la miniaturisation, l'ordinateur n'a plus une activité seulement centrée sur son utilisateur final, l'informatique prend aussi en compte l'environnement grâce à des informations issues de capteurs qui peuvent communiquer entre eux. L'environnement peut être très divers : la maison, la voiture, le bureau, la salle de réunions.

Avant le développement du concept *intelligence ambiante*, le terme *informatique ubiquitaire* a été proposé par Weiser (1994) dans le projet Xerox PARC pour désigner un modèle d'interaction homme-machine dans lequel le traitement de l'information relatif aux activités de vie quotidienne a été intégré dans les objets. La vision de l'informatique ubiquitaire a été permise par l'utilisation de très nombreux dispositifs miniaturisés, économiques, robustes et connectés par réseau que l'on a pu dès lors installer dans des endroits où les personnes réalisent leurs activités quotidiennes. Ensuite, Uwe Hansmann de la société IBM a proposé le terme *informatique pervasive* pour décrire cette technologie (Nic, 2003). Dans la littérature, il existe une grande confusion concernant l'utilisation de ces deux termes et celui d'intelligence ambiante. Ils sont souvent utilisés de façon indistincte (Ronzani, 2009). Pourtant, nous considérons plus pertinente la distinction proposée par (Augusto, 2009) qui affirme que le concept d'intelligence ambiante recouvre un plus large domaine que la simple disponibilité omniprésente de ressources et qu'elle nécessite l'application de l'intelligence artificielle pour atteindre son but, à savoir avoir un comportement judicieux et faire preuve d'initiative. Ainsi, à la différence des deux autres concepts, l'intelligence ambiante fait appel à des contributions d'autres domaines comme l'apprentissage automatique, les systèmes multi-agents

ou la robotique.

Le terme *intelligence ambiante* a été proposé lors d'un atelier organisé par Philips dont le but était le développement de scénarios pour l'utilisation de dispositifs fournissant des information et communication ubiquitaires (Zelkha, 1998). L'ubiquité, une caractéristique de l'intelligence ambiante, est la propriété d'un système de permettre à l'utilisateur une interaction à partir de n'importe quel point de l'environnement qu'il contrôle. Les progrès en Traitement de l'Information qui ont permit le développement de cette discipline concernent surtout les réseaux de capteurs, les environnements perceptifs et l'intelligence artificielle. Actuellement, les capteurs domotiques et les unités de traitement deviennent très répandus, ils sont connectés à des réseaux domotiques et contrôlés par logiciels spécialisés. Ces capteurs peuvent être enfouis facilement dans l'habitat et des objets d'utilisation quotidienne tels que la télévision, le chauffage, et les meubles. Même les espaces publics sont de plus en plus équipés de capteurs de surveillance pour assurer la sécurité ou la planification. Les avancées récentes dans les méthodes d'intelligence artificielle ont donné lieu à plusieurs applications qui exploitent les informations, souvent incertaines et imprécises, obtenus des environnements perceptifs comme par exemple la modélisation et l'apprentissage automatique des situations, le raisonnement temporel pour le traitement des événements dans des environnements perceptifs, ou l'analyse du comportement de l'habitant (Mozer, 2005; Chua et coll., 2009). La représentation de connaissances, un problème très abordé dans l'intelligence artificielle, présente une importance majeure pour l'intelligence ambiante. La figure 2.1 montre les relations existant entre l'intelligence ambiante et l'intelligence artificielle.

Augusto et coll. (2010) ont identifié les taches les plus importantes pour un système d'intelligence ambiante :

(a) l'apprentissage des habitudes, préférences, et besoins des utilisateurs, avec en plus l'adaptation des modèles au cours du temps,

(b) le diagnostic correct des situations,

(c) le suivi de l'occurrence des événements,

(d) l'intégration des dispositifs périphériques mobiles,

(e) une façon structurée d'analyser, de décider, et de réagir sur l'environnement.

2.1.1 Environnements intelligents

Un environnement intelligent est issu du concept d'intelligence ambiante, il consiste en l'implémentation d'un espace physique dynamique et adaptable qui optimise les services aux utilisateurs en utilisant des systèmes sensibles au contexte (cf. 2.2.2) et des technologies ubiquitaires. L'accès ubiquitaire est rendu possible par la présence de multiples systèmes hétérogènes connectés par un réseau et incorporés dans le milieu (meubles, vêtements, murs) et qui sont capables de détecter l'apparition de phénomènes physiques et éventuellement de les caractériser. Selon Cook et Das (2005), le terme *intelligent* fait référence à la capacité d'acquérir et d'appliquer des connaissances de façon autonome, tandis que le terme *environement* fait référence au milieu dans lequel demeure l'utilisateur.

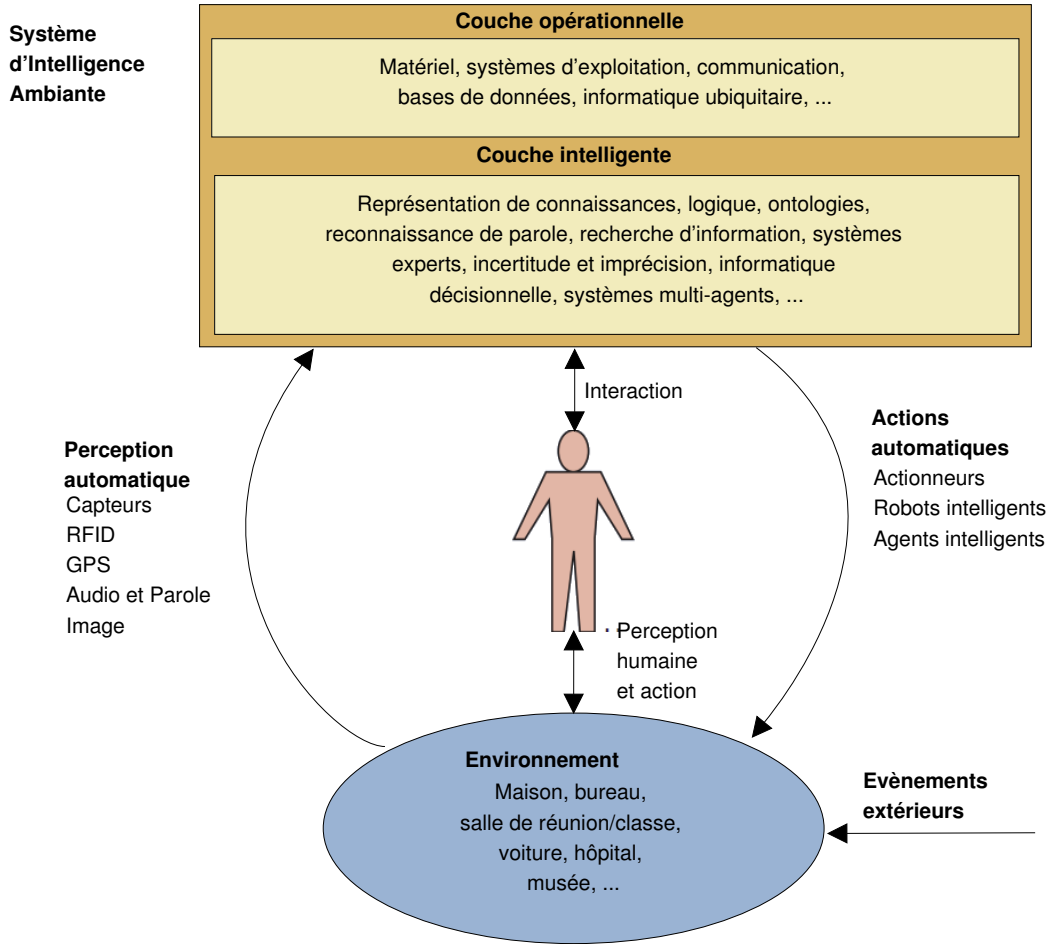


FIGURE 2.1: Relation entre l'intelligence ambiante et l'intelligence artificielle selon Ramos et coll. (2008)

Les capteurs sont les éléments nécessaires pour faire le lien entre le monde physique et le monde virtuel, le système intelligent effectue ses analyses dans le monde virtuel. Il est aussi nécessaire de disposer d'actionneurs pour que le système puisse agir concrètement sur l'environnement perceptif. Il convient de noter cependant que le contrôle des actionneurs ne permettra de satisfaire les besoins des utilisateurs que si le système logiciel implémente des méthodes de l'intelligence artificielle. Si nous considérons l'exemple d'une voiture, elle peut être considérée comme un environnement intelligent si elle comporte un système capable d'acquérir des informations suffisamment précises sur les trajets fréquents effectués par un conducteur (utilisation des données GPS par exemple), sur l'état de l'encombrement des routes (parfois disponible sur internet), et sur la disponibilité du carburant ; ces informations sont nécessaires pour déterminer les trajets optimums permettant de diminuer le temps de parcours ou de recommander l'approvisionnement de carburant dans une station service précise et éviter une panne sèche.

Les environnements perceptifs concernent des lieux de vie, ils affectent donc directement la vie des personnes. C'est la cas par exemple des étudiants dans une salle de classe, des habitants dans un appartement, des piétons dans la rue, ou des clients dans un magasin. Ils verront leurs modes de vie influencés par les technologies installées en ces lieux. Si les sys-

tèmes fonctionnaient de façon idéale, la vie des utilisateurs devrait s'en trouver améliorée. Cependant, ce n'est pas toujours le cas car les systèmes informatiques sont souvent affectés par plusieurs facteurs externes au domaine du système comme l'incertitude de l'information ou les erreurs d'observations des capteurs utilisés.

Étant donné l'utilisation de l'intelligence ambiante dans des lieux de vie, il est primordial de considérer les aspects étiques que cela induit. Ces aspects concernent principalement :

Respect de la vie privée. Les caméras vidéo, par exemple, peuvent être utilisées pour surveiller les rues et détecter des actes criminels ou les magasins pour prévenir les vols. Elles peuvent être utilisées dans les maisons intelligentes pour détecter des situations de risque, ce qui est intéressant par exemple pour aider les personnes en perte d'autonomie. Pourtant, dans de nombreux cas, les caméras de vidéo sont mal acceptées, voire rejetées par les utilisateurs. L'acceptation de ces dispositifs dépend beaucoup des valeurs culturelles de la personne et de la situation elle-même. Quelques utilisateurs peuvent accepter de perdre une partie de leur intimité si cela permet de leur garantir une meilleure sécurité, tandis que d'autres n'accepteraient jamais des dispositifs qui exposent des détails intimes de leur vie quotidienne. Ces aspect concernant l'acceptabilité de la surveillance ont été étudiés par Rumeau et coll. (2006).

Sécurité de l'information Les capteurs obtiennent des informations concernant les activités de vie quotidienne et il existe des méthodes pour trouver des motifs de comportement en exploitant les données enregistrées. Il est possible que ces informations relatives aux habitudes, aux maladies, à la situation des personnes soient utilisées par des individus non autorisés désirant en tirer un profit illicite. Les implémentations des systèmes perceptifs doivent donc garantir aux utilisateurs que leurs données ne seront pas utilisées pour des buts non souhaités.

Dernièrement, l'attention a été focalisée sur l'implémentation des interfaces naturels homme-machine dans les environnements intelligents, c'est-à-dire sur des interfaces invisibles qui utilisent, pour l'interaction, des éléments naturels comme les gestes, le mouvement du corps ou la voix. Une des utilisations la plus répandue de la technologie des environnements intelligents est la domotique qui consiste à automatiser les différents dispositifs d'un bâtiment (système de chauffage, volets roulants, porte de garage, portail d'entrée, prises électriques, etc.).

2.1.2 Domotique

La domotique regroupe l'ensemble des technologies permettant l'automatisation des équipements d'un habitat. Ce terme vient du mot latin *domus* qui désigne une demeure patricienne. La domotique vise à apporter des fonctions de confort (commandes à distance, gestion d'énergie, optimisation de l'éclairage et du chauffage, etc.), de sécurité (alarme) et de communication (contacts et discussion avec des personnes extérieures). Il existe plusieurs points en commun entre la domotique et l'intelligence ambiante, par contre la première se centre dans les bâtiments intelligents (par exemple, la maison, le bureau, la chambre d'hô-

tel) et l'autre couvre une gamme d'applications plus vaste, elle peut concerner la circulation des véhicules ou les espaces publics (une station de métro, un parc d'agrément, une autoroute). En fait, l'intelligence ambiante ne s'occupe pas seulement de l'automatisation mais aussi entre autres de la surveillance, de l'analyse de comportement, ou de la présentation d' information en contexte. Une autre différence importante est que la domotique ne nécessite pas forcement l'utilisation des techniques de l'Intelligence Artificielle (IA). Par exemple, pour améliorer le confort et l'économie de l'énergie, une solution domotique simplifiée ne demande pas d'utiliser l'IA si elle consiste seulement à allumer les lumières d'une pièce lorsque la personne est détectée par un capteur de présence. Malgré tout, l'application de l'intelligence ambiante est très prometteuse pour améliorer les services, et l'objectif de nos travaux de recherche vise dans ce sens à améliorer le confort et la sécurité.

Les fonctions qui sont couvertes par la domotique sont classées en trois catégories selon Friedewald et coll. (2005) :

- **Support aux fonctions basiques du logement**. Ces fonctions sont directement liées au confort et à l'optimisation de la consommation de ressources énergétiques. Il s'agit de faciliter ou automatiser le contrôle des fonctions basiques tel que le chauffage, la ventilation, la lumière, ou l'utilisation des appareils électroniques. L'intelligence ambiante peut être appliquée pour l'implémentation des environnements intelligents capables de contrôler ces fonctions par l'analyse des ordres vocaux ou les gestes, par exemple. L'apprentissage automatique du comportement de l'habitant pour améliorer son confort est une autre possible application.
- **Sécurité.** La sécurité des personnes et des biens matériels est un des enjeux des produits utilisant les technologies de l'information. Les fonctions de sécurité peuvent être liées à l'accès physique à l'environnement (systèmes anti-cambriolage), ou la santé (prévention et surveillance). La reconnaissance de vidéo est souvent à la base des solutions proposées.
- **Support à l'autonomie.** Dans un contexte de vieillissement de la population, l'objectif est le développement d'outils pour améliorer l'autonomie et l'indépendance des personnes fragiles, handicapées ou âgées. Ces outils doivent fournir, en plus d'une surveillance, des moyens pour compenser des troubles éventuels (déficience auditive ou visuelle, par exemple) et améliorer la communication avec l'extérieur.

2.1.3 Maisons intelligentes

La maison intelligente est le domaine applicatif dans lequel nous menons nos travaux, c'est un domaine qui a attiré beaucoup d'attention dans la communauté de l'intelligence ambiante en partie à cause des applications possibles dans le maintien à domicile des personnes âgées, sujet que nous aborderons plus en détail par la suite, mais aussi dans l'optimisation de la consommation énergétique. Une *maison intelligente* ou *smart home* (Aldrich, 2003; Chan et coll., 2008) est une résidence équipée de technologie d'intelligence ambiante, qui anticipe et répond aux besoins de ses occupants en essayant de gérer de manière op-

timale leur confort et leur sécurité par action sur la maison, et en mettant en œuvre des connexions avec le monde extérieur.

Augusto et Nugent (2006) considèrent la maison intelligente comme un système dynamique qui débute par un état initial et puis se déroule en parcourant différent états en fonction des événements qui ont lieu dans l'environnement. Chaque état est défini par la valeur courante des variables du système, comme la localisation de la personne et l'état d'activation des dispositifs. Ces valeurs sont discrètes, par exemple *arrêt/marche* ou continues comme la température. L'analyse temporelle dans une maison intelligente consiste à évaluer l'information d'une séquence d'événements passés dans une fenêtre temporelle pour inférer avec plus de précision la situation courante de l'habitant ou prédire la configuration possible de l'état suivant. L'évolution du système passe à travers une séquence d'états S_i qui sont produits comme résultat des états E_i qui ont été enregistrés : $S_0, E_1, S_1, E_2, S2$.... Considérons l'exemple de l'évolution de l'état de différents capteurs CME, CPC et CMC dans une maison intelligente affichée sur la figure 2.2. CME représente un capteur de mouvement dans l'entrée de la maison, CPC un contact de porte dans la cuisine, CMC un capteur de mouvementent dans la cuisine. Les états $n+1...n+3$ correspondent aux événements correspondant à l'entrée de la personne dans la maison, à son déplacement vers la cuisine et à la mise en route d'une plaque électrique.

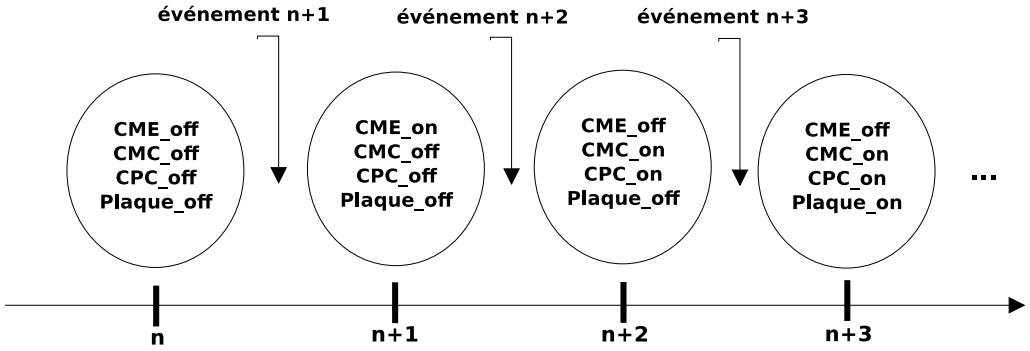


FIGURE 2.2: Exemple de système dynamique d'une maison intelligente

Les objectifs les plus importants, que l'on cherche à atteindre en élaborant une maison intelligente, sont les suivants :

- Aider les habitants à conserver leur santé, leur sécurité et leur bien être.
- Réaliser plusieurs taches automatiquement pour soulager l'habitant de l'attention nécessaire au contrôle des dispositifs de la maison.
- Intégrer les activités domestiques, de travail et de loisir dans la maison.
- Occulter les détails techniques pour que l'habitant n'ait pas besoin de connaître leurs fonctionnement.

L'amélioration de la productivité est un autre argument en faveur de l'implémentation des maisons intelligentes. Un nombre croissant de personnes sont concernées par le télétravail. Cependant, les aspirations, les besoins et les désirs des personnes vis à vis de la maison intelligente sont extrêmement complexes, ceci explique le nombre important d'études qui y

sont consacrées actuellement.

Friedewald et coll. (2005) affirme que la maison ne peut pas être une machine ni un ensemble d'outils destinés à seulement aider les habitants dans leurs tâches de la vie quotidienne. La maison doit être un espace pour les humains, leur qualité de vie et la perception qu'ils en ont doit être améliorée par la technologie et l'intelligence ambiante. La maison n'est pas seulement un espace physique mais aussi un espace de vie socio-culturel.

Un phénomène social qui a motivé la recherche dans les maisons intelligentes dernièrement est le vieillissement rapide de la population dû à l'amélioration des techniques médicales et chirurgicales. Le nombre croissant des personnes âgées qui souffrent d'une perte d'autonomie pose un problème à cause d'une part du nombre limité de places dans les institutions spécialisées, mais aussi d'autre part du coût élevé du séjour dans ces institutions. Il existe de nombreux travaux dans le domaine des technologies d'information s'intéressant aux technologies pour la santé des personnes âgées (Micera et coll., 2008). Le but de ces recherches est de mettre au point des techniques efficaces d'assistance à domicile et de conception de l'habitat pour permettre aux personnes âgées d'y vivre de manière autonome, en bonne santé, et dans le confort et la sécurité. Les applications de l'intelligence ambiante concernent, entre autres, la conception de bâtiments adaptés aux personnes âgées(Raad et Yang, 2009), la communication avec les personnes de leur entourage, et l'assistance à domicile (avec des détections de situations anormales) (Romdhane et coll., 2011).

Le bâtiment intelligent est souvent au centre des projets européens liés à l'Assistance à la Vie Autonome (AVA), ou *Ambient Assisted Living (AAL)*, étant donné qu'il répond à certains besoins d'assistance médicale (Haigh et Yanco, 2002; Lacombe et coll., 2005; Patterson et coll., 2002; Tang et Venables, 2000; Rialle, 2007). En effet, l'assistance aux personnes en perte d'autonomie est un objectif prioritaire de la commission européenne (6e et 7e PCRD). Parmi ces projets internationaux, on peut citer RoboCare (Bahadori et coll., 2004), CompanionAble (Badii et Boudy, 2009) et COGNIRON [1] (Li et Wrede, 2007) qui utilisent un robot assistant communiquant avec la maison intelligente, ALADIN (Maier et Kempter, 2010) qui étudie l'effet de la lumière sur le bien être, ou encore EMERGE [2] qui vise à détecter quand les personnes s'écartent de leurs habitudes de vie.

2.1.4 Projets de maisons intelligentes

Le nombre de projets liés au bâtiment intelligent est devenu tellement important qu'il devient difficile d'en faire un tour complet ; c'est pourquoi nous nous limiterons à la présentation de quelques projets qui nous semblent les plus pertinents et en rapport avec notre étude.

1. http://www.cogniron.org/final/Home.php
2. http://www.emerge-project.eu/

2.1.4.1 House_n - Massachusetts Institute of Technology (MIT)

Le projet House_n[3] (Intille, 2002) couvre les domaines de l'informatique, de l'architecture, de la mécanique et de la médecine préventive. Il est dirigé par des chercheurs du département d'architecture du MIT qui s'occupent de la conception de la maison, des technologies et des services à intégrer afin que ceux-ci puissent évoluer pour faciliter les principales activités de la vie quotidienne, telles que la communication avec l'extérieur, l'assistance médicale, et, puissent améliorer l'autonomie des personnes âgés (Kent et coll., 2003).Les objectifs principaux du projet House_n sont :

- l'application de techniques d'interaction homme-machine pour encourager des comportements sains (régime alimentaire, exercices...) ;
- la reconnaissance des activités — surtout pour les personnes âgés — ;
- la surveillance biométrique à travers des dispositifs mesurant le rythme cardiaque, la tension artérielle, la respiration et le taux de glycémie.

Une habitation, le *Place Lab*, a été complètement équipée afin d'étudier le comportement des personnes et leurs interactions avec les objets et l'environnement de l'habitat. Des volontaires ont acceptés de vivre à l'intérieur pendant une période de quelques jours à une semaine. Une particularité de ce projet est le nombre impressionnant de capteurs utilisés pour enregistrer les activités développées (plusieurs centaines). Les principaux capteurs utilisés sont :

- des capteurs d'état sans fil placés sur les objets utilisés ou manipulés par les personnes, il s'agit notamment de contacts positionnés au niveau des portes, des fenêtres, des tiroirs de commode ainsi que dans les récipients de la cuisine ;
- des dispositifs radio fréquence pour localiser les habitants ;
- des microphones pour capturer l'information audio ;
- un système de capture vidéo qui inclut des caméras infrarouges ;
- des dispositifs PocketPC pour recevoir les retours de l'utilisateur ;
- et des capteurs embarqués maintenus sur le corps du volontaire pour la surveillance biométrique.

Il est important de noter que les volontaires ne sont pas spécifiquement des personnes âgées ou handicapées mais des personnes de la population active.

Une des préoccupations majeures du projet est la reconnaissance d'activité. Les membres du projet préconisent une approche ayant pour base un réseau bayésien (Tapia et coll., 2004) et affirment que l'utilisation d'un modèle dynamique tel qu'un HMM(voir 4.2.1.3) ne serait pas pertinent dans un environnement multicapteurs. En fait, le projet consiste surtout à tester de nouveaux capteurs et de nouvelles interfaces ainsi qu'à recueillir des données pour analyser le comportement de la personne. Le projet ne fournit pas, à notre connaissance, d'architecture logicielle globale pour permettre à l'habitation de capturer l'environnement, pour prendre des décisions, et pour agir sur l'environnement.

3. http://architecture.mit.edu/house_n/

2.1.4.2 CASAS - Washinton State University

Dans le projet CASAS [4], l'habitat est vu comme un agent intelligent qui perçoit son environnement à partir d'un ensemble de capteurs et réagit à travers des dispositifs capables de piloter le chauffage ou la lumière. Le but principal est de mettre en œuvre un contrôleur intelligent capable de raisonner pour minimiser le coût de fonctionnement de l'habitat et en même temps d'améliorer le confort des personnes. Les expérimentations sont effectuées dans un appartement situé sur le campus de l'université de Washington, appartement qui comporte une salle de séjour, une cuisine, des toilettes et trois chambres. Très souvent, les projets de maisons intelligentes définissent *a priori* un ensemble d'activités à reconnaître ; au contraire, dans le projet CASAS, les habitudes de l'habitant sont apprises sans connaissance *a priori* (Rashidi et coll., 2011). Des techniques d'apprentissage de règles d'association sont utilisées pour apprendre les motifs fréquents et périodiques à partir de données reçues par les capteurs. De plus, les membres du projet présentent des algorithmes permettant de s'adapter aux changements de motifs et d'utiliser le retour de l'utilisateur comme entrée. L'apprentissage de motifs en utilisant une extension des règles temporelles définis par Allen (Allen et Ferguson, 1994) a été effectué dans une autre étude (Jakkula et coll., 2007) afin de représenter l'ordre des événements dans une activité.

L'efficacité de l'utilisation des ressources a aussi été traité dans ce projet (Chen et Cook, 2012). Par ailleurs, les données des capteurs ont été exploitées afin d'apprendre des modèles de régression pour prédire la consommation électrique. Le propos de cette application est d'aider l'utilisateur à réaliser ses activités quotidiennes de manière à diminuer la consommation énergétique.

Les différents composants de l'environnement intelligent envisagé dans ce projet sont représentés dans la figure 2.3. Les capteurs sont connectés au système d'exploitation via des interfaces matérielles dédiées. Une couche communication permet de faire le pont soit entre les dispositifs matériels eux-mêmes, soit entre les dispositifs matériels et les services logiciels. Le système de décision peut être fondé sur un système expert mais peut aussi utiliser des modèles probabilistes. La connaissance est externalisée dans une couche spécifique. L'architecture présentée est rudimentaire et reste une perspective du projet qui affirme se concentrer surtout sur la fouille de données et la modélisation. Elle est cependant très proche de celles de projets plus avancés.

2.1.4.3 The Intelligent Dormitory (iSpace) - The University of Essex

ISpace [5] (Holmes et coll., 2002) est un environnement dédié à l'intelligence ambiante situé sur le campus de l'Université d'Essex. Cet espace est constitué d'un appartement type F2 avec une organisation similaire à celle des chambres des résidences d'étudiants de cette université ; pour les besoins du projet, il est équipé de dispositifs destinés à enregistrer des données relatives à l'activité de l'occupant afin d'apprendre son comportement. Des action-

4. http://ailab.wsu.edu/casas/
5. http://cswww.essex.ac.uk/iieg/idorm.htm

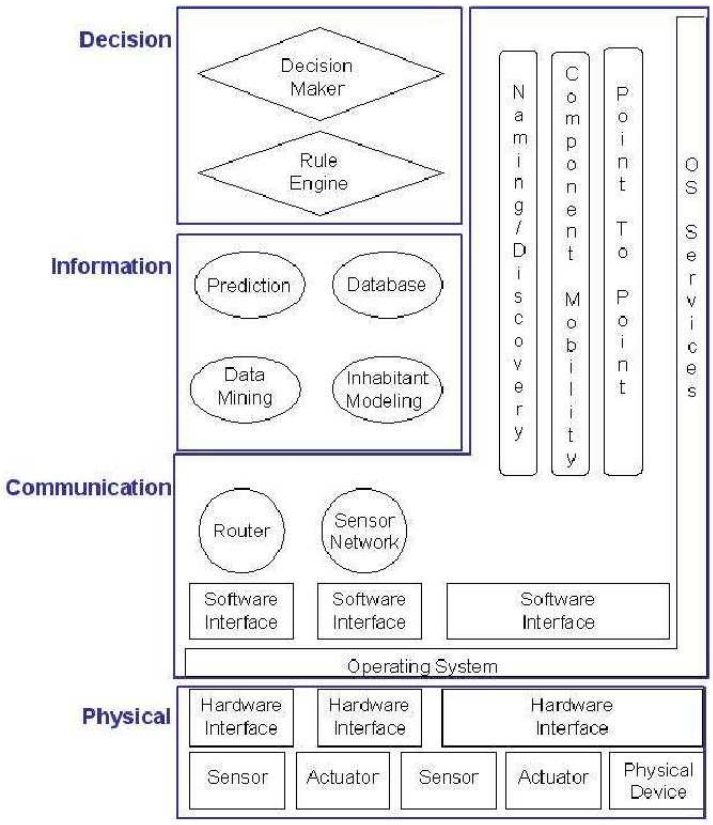


FIGURE 2.3: Environnement intelligent du projet CASAS (Cook et Das, 2007)

neurs destinés au pilotage du chauffage ou de la lumière peuvent être contrôlés à travers plusieurs interfaces : PDA (*Personal Digital Assistant*), ordinateurs connectés au réseau ou téléphones portables connectés par WAP (*Wireless Application Protocol*). Comme dans le projet CASAS, l'objectif principal est d'étudier les méthodes d'apprentissage et leur adaptation, le but final étant l'étude des mécanismes et outils pour mettre en œuvre un environnement qui puisse apprendre de façon non intrusive les habitudes des personnes et les aider dans leur vie quotidienne. L'architecture choisie pour mettre en œuvre ce système repose sur les technologies de communication réseau et la distribution d'agents pour contrôler les dispositifs de la chambre. La conception de ces agents est principalement à base de logique floue (Doctor et coll., 2005). L'application développée pour coordonner des dispositifs hétérogènes à travers les agents intelligents se base sur une optimisation multi-objectif (minimisation de la charge sur le réseau, optimisation du confort de l'utilisateur, et minimisation de la consommation d'énergie) (Tawil et Hagras, 2005).

2.1.4.4 GER'HOME - CSTB

GER'HOME [6] (Zouba et coll., 2009) est un projet du Centre Scientifique et Technique du Bâtiment (CSTB) dont le but est de concevoir, d'expérimenter et de certifier des solutions techniques supportant des services d'aide au maintien à domicile des personnes âgées, en utilisant des technologies domotiques intelligentes pour assurer autonomie, confort de vie,

6. http://gerhome.cstb.fr/

sécurité, surveillance et assistance à domicile. Un local est utilisé comme appartement de test, il est composé d'un couloir, de 2 pièces et d'une salle de bain et est entièrement aménagé pour permettre son utilisation en logement. Dans la partie technique de ce projet, l'accent est mis sur la reconnaissance des Activités de la Vie Quotidienne (AVQ) ou *Activities of Daily Living* (ADL) (Katz, 1983) et la reconnaissance de situations à partir de l'analyse de vidéos et de trace de capteurs binaire dans l'appartement dédié (contacts de porte, débitmètre...).

La reconnaissance des activités a été fait par une méthode de *Bag of words* en prenant en compte la relation spatio-temporelle des attributs(Bilinski et Brémond, 2012). D'ailleurs, un cadre de travail pour la reconnaissance d'activité basé sur la théorie des observations datées (Goc, 2006) a été proposé par Pomponio et coll. (2011). Dans ce travail, les activités sont considérées comme des entités qui puissent être représentées à différents niveaux d'abstraction.

2.1.4.5 Aging In Place - University of Missouri

Aging in Place[7] (Skubic et coll., 2009) est un projet qui a pour but de donner aux personnes âgées un espace indépendant qui permette de garantir une surveillance médicale adéquate grâce à des capteurs dédiés, mais qui soit confortable de manière à ce que les habitants se sentent chez eux. Un bâtiment nommé *Tiger Place* (Rantz et coll., 2008) a été construit dans le respect des normes sanitaires pour faire des expérimentations avec des volontaires. Ces personnes ont vécu dans cet espace pendant des périodes variants entre 3 mois et 3 ans, avec une moyenne de 15 mois. Parmi les volontaires, ont trouve des personnes avec des maladies chroniques telles que l'arthrite, le diabète, des troubles cardiaques ainsi que des stades initiaux de la maladie d'Alzheimer. Un large éventail de professionnels, infirmiers, thérapeutes, physiologistes ou spécialistes de la conception environnementale, ont participé à la création de ce bâtiment qui contient 31 appartements indépendants, mais aussi des espaces communs pour l'exercice physique et les loisirs. Chaque appartement est composé d'une cuisine, de toilettes, d'une chambre, d'une salle à manger et d'un séjour. L'originalité de ce projet est qu'il ne s'agit pas seulement d'installations ponctuelles mais de véritables logements qui ont été occupés par de véritable occupants pendant des périodes très longues.

Le principal objectif de *Tiger Place* est la détection de situations de détresse à travers l'analyse des données fournies par les capteurs (Alexander et coll., 2011). Le système mis en œuvre s'appelle *In-Home Monitoring System* (IMS), il comprend des dispositifs tels que des capteurs de mouvement et des contacts de porte (sur le placard et le frigo). Pour permettre un plus grand confort et éviter une trop grande intrusion, les dispositifs fixés sur la personne ont été évités.

La reconnaissance vidéo a été utilisée dans le but de compléter les informations des autres capteurs notamment pour la détection des événements anormaux, tels que la chute (Wang et coll., 2011), mais aussi d'éviter les fausses alarmes. Le problème concernant le re-

7. urlhttp ://aginginplace.missouri.edu/

spect de la vie privé à été contourné en faisant une séparation entre le contour de l'habitant (qui apparait seulement comme un silhouette blanche) et l'arrière plan. Ils affirment que, bien que les personnes âgées ne soient pas à l'aise avec l'utilisation de cette technologie dans leur appartement, elles acceptent plus facilement l'utilisation de ce type d'images où seule des silhouettes grossières sont enregistrées(Xi et coll., 2006). Des méthodes basées sur la logique floue ont été utilisées pour l'extraction de ces silhouettes.

2.1.4.6 CompanionAble - UE

CompanionAble [8] est un projet financé par la communauté européenne de 2008 à 2011, son but est l'aide la stimulation cognitive et la gestion de thérapie des personnes suivies à domicile. Ce support est donné à travers un « compagnon » robotique (mobile) fonctionnant de manière collaborative avec l'environnement informatique ambiant de l'habitat (stationnaire). Ce robot de taille importante est apte à se déplacer à l'intérieur du domicile pour porter assistance à la personne. Le but est de concevoir un système d'assistance ayant pour base l'observation multimodale et l'interaction homme-machine. Ce projet s'intéresse en particulier à l'interprétation des émotions à travers l'analyse vidéo (Stricker et coll., 2010). Une originalité de ce projet est d'inclure une interface audio pour la communication entre le robot et l'habitat, le robot étant capable de se déplacer et donc de s'approcher à une distance raisonnable de la personne (Milhorat et coll., 2012).

2.1.4.7 SOPRANO - UE

SOPRANO (*Service-Oriented Programmable Smart Environments for Older Europeans*) [9] (Wolf et coll., 2008) est un projet financé par la Commission Européenne qui vise à aider les personnes âgées à avoir une vie plus indépendante. L'apport principal du projet a été la création de l'intergiciel (*middleware*) OpenAAL pour la mise en œuvre logicielle des maisons intelligentes. Il s'agit d'une infrastructure logicielle ouverte qui prend comme entrée les informations provenant des capteurs installés dans la maison et qui active des services pour effectuer des actions sur l'environnement. La figure 2.4 montre l'architecture de OpenAAL. Celle-ci est orientée vers les services (*Service Oriented Architecture*, SOA) et est fondée sur OSGi. L'originalité du système est de fournir deux couches pour les représentations sémantiques de l'environnement : la première est une description des capteurs et de leurs états et, la deuxième décrit la situation de l'habitant, ses activités, sa localisation, les urgences.

L'interface au niveau capteurs permet d'inclure facilement des capteurs grâce à un protocole de communication entre un capteur et l'interface. L'information fournie par les capteurs est décrite par une ontologie « métier » spécifique aux capteurs qui fournit un vocabulaire commun pour l'échange des données. Ainsi, les fournisseurs de capteurs peuvent intégrer leurs composants indépendamment des autres composants du système en respectant cette description. L'interface de service est mise en œuvre à un plus haut niveau. Elle permet aux

8. http://www.companionable.net/
9. http://www.soprano-ip.org/

services développés sur SOPRANO de faire des requêtes sur l'information contextuelle, telle que la localisation de l'habitant. Ce projet mettant l'accent sur l'utilisation des ontologies, la reconnaissance de situations est réalisée par le traitement des évènements complexes ayant une description sémantique(Xu et coll., 2010, 2011). L'approche suivie permet de définir, à travers de la logique de description, des évènements complexes capables d'exprimer des relations spatio-temporelles et qui décrivent les situations à reconnaître.

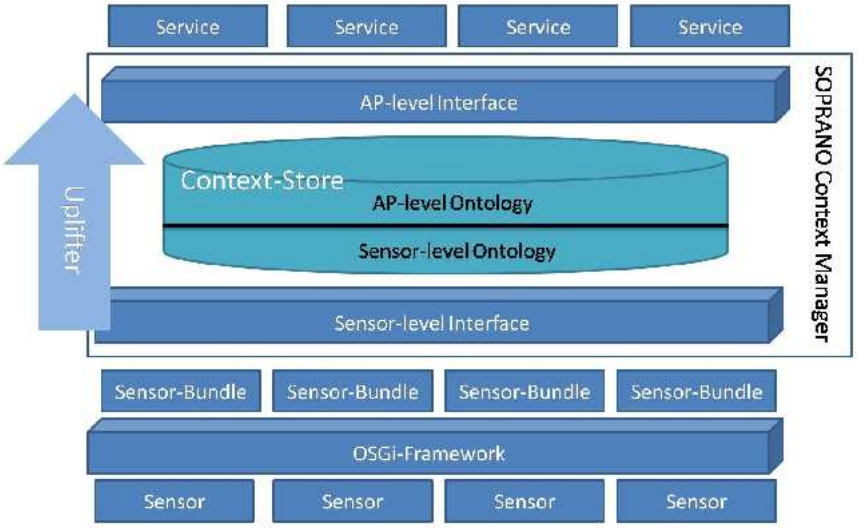


FIGURE 2.4: Architecture de l'intergiciel OpenAAL

Les membres du projet SOPRANO ont eu la volonté de développer un logiciel libre, OpenAAL, qui puisse être facilement installé et utilisé par d'autres fournisseurs de technologies. Wolf et al (Wolf et coll., 2008) expliquent comment les personnes concernées par la mise en œuvre d'une maison intelligente peuvent travailler avec OpenAAL. Quatre groupes indépendants ont été identifiés :

(1) les fournisseurs de capteurs qui peuvent inclure leurs dispositifs en suivant les spécifications de l'ontologie ;

(2) les développeurs de logiciel qui implémentent des algorithmes de raisonnement pour faire l'inférence de contexte ;

(3) les ingénieurs de service d'aide, des professionnels sans grande connaissance technique, qui peuvent utiliser les différents outils sous forme de services pour, par exemple, personnaliser le contrôleur intelligent selon les besoins de l'habitant et trouver une configuration qui réduise les coûts ;

(4) et les donneurs de soins, qui personnalisent le contrôleur selon les conditions d'une maison spécifique, cela peut aussi être l'habitant lui-même.

2.1.4.8 Smart hoMes for All / SM4ALL - UE

À l'instar de SOPRANO, le projet SM4ALL (Catarci et coll., 2009), financé par la commission européenne, a comme but la construction d'une plate-forme pour le développement

des environnements perceptifs. Il prévoit la réalisation d'une plateforme dédiée permettant l'implémentation de la collaboration de services dans les environnements perceptifs. Cette collaboration est réalisée grâce au traitement des inter-relations entre les composants d'un système (la composabilité), et à l'application de technologies sémantiques pour la configuration automatique des services. Les propriétés que devra garantir cette plate-forme sont la modularité par rapport au nombre de services à contrôler et le comportement dynamique pour gérer au fil de l'eau l'insertion ou la suppression des dispositifs ou des applications, mais aussi la découverte automatique de services (Rasch et coll., 2011).

La reconnaissance de situation est implémentée à partir de l'analyse du comportement de l'habitant et l'état courant du système. Dans le cas où le système reconnait une situation dans laquelle une aide est nécessaire pour que la personne puisse terminer une activité, il propose une liste d'options nécessitant la collaboration de plusieurs services installés sur la plateforme. La coordination de services est donc très importante dans ce projet, d'où la proposition d'une technique de spécification d'orchestration qui aide les services à collaborer entre eux. Il s'agit d'un système sensible au contexte car les options offertes prennent en considération le contexte courant de l'utilisateur. Pourtant, dans ce projet la prise de décision automatique n'a pas été implémentée.

L'architecture de ce projet (Kaldeli et coll., 2010) conserve une similarité avec SOPRANO : la division en couches pour décrire les éléments physiques et les services. Dans SM4ALL, il existe une première couche appelée *pervasive* qui groupe et décrit les dispositifs installés dans l'environnement. Une deuxième couche appelée *composition* a pour rôle l'organisation de services, ce qui nécessite des mécanismes d'inférence basés sur des règles logiques. À la différence de SOPRANO, les services sont rendus à l'utilisateur suivant une approche dynamique. Ainsi, les actions à exécuter pour satisfaire les besoins de l'utilisateur sont organisées dynamiquement selon les services disponibles et les préférences de l'utilisateur.

Un aspect original de SM4ALL est l'utilisation d'une interface homme-machine fondée sur l'analyse de l'activité neuronale (Aloise et coll., 2011). Cela peut être utile pour les personnes ayant un handicap moteur. Il s'agit d'une technique qui détecte automatiquement les intentions de l'utilisateur selon des motifs reconnus dans les ondes cérébrales détectées au moyen d'électrodes placées sur le crâne du patient. Une interface graphique affiche un menu avec plusieurs icônes et les signaux cérébraux sont utilisés pour déterminer l'icône sur laquelle se concentre l'utilisateur. Après reconnaissance de l'intention, une commande est envoyée pour exécuter l'ordre domotique.

2.1.4.9 HomeCare - LAAS Toulouse

Ce projet est en fait la poursuite d'un travail mené depuis plusieurs années par le Laboratoire d'Analyse et d'Architecture de Systèmes à Toulouse (LAAS), les projets « Prosafe » puis « Ourses » (Offre d'usage rural de service par satellite), permettant le suivi des personnes âgées. L'objectif était la mise en place d'un service d'assistance médicale à distance, de surveillance comportementale et de détection de situations anormales : chute, malaise, fugue, agitation nocturne ». Le projet HomeCare se centre sur la surveillance des personnes âgées atteintes de

la maladie d'Alzheimer dans les établissements spécialisés (Charlon et coll., 2011). Le projet utilise une unité de soins de l'hôpital de Caussade à Toulouse destinée à ce type de patients et composée de douze appartements individuels équipés de capteurs, une salle de vie collective et un local réservé au personnel soignant. Cette plate-forme expérimentale est équipée d'un réseaux de capteurs sans fils. Les capteurs utilisés sont des capteurs de mouvement infrarouges, des capteurs de présence pour les lits, et une étiquette électronique que doit être toujours portée par la personne et sert à la localisation de la personne et à la détection des chutes (capteur de mouvement et balise Zigbee). Les capteurs sont déployés dans les chambres des patients, le couloir, la salle de vie, la terrasse et le jardin. Les objectifs principaux du projet sont doubles, tout d'abord la détection en temps réel des situations de danger (chute, immobilité prolongée, fugue, etc.), mais aussi la reconnaissance des déviances dans le comportement par suivi des activités. L'approche retenue est basée sur l'apprentissage automatique pour apprendre les modèles de comportement avec des réseaux de neurones à partir de l'historique des événements recueillis au cours des 30 derniers jours. Lorsque il y a dépassement de seuils déterminé préalablement, une alarme se déclenche. Les paramètres observés sont le taux de mobilité moyen, l'agitation nocturne moyenne, le temps de séjour moyen et les accélérations qui permettent de détecter les chutes.

La fonction d'identification permet de surveiller les patients 24 heures sur 24. Les résultats de la fonction de détection de chute sont encourageants, 7 chutes sur 8 ont été détectées et 1 fausse alarme a été observée par semaine en moyenne. La fonction de localisation permet au personnel médical d'intervenir immédiatement en cas de situation de danger (alerte par teléphone). La mesure de l'actimétrie a permis au personnel médical de mieux observer l'effet d'un traitement pour l'un des patients suivis.

2.1.4.10 Positionnement de nos travaux par rapport à ces projets

Tout d'abord, on pourra remarquer que les projets ne s'orientant pas exclusivement sur une application médicale et réutilisant des normes établies sont ceux qui apportent les solutions les plus génériques et complètes. On peut prendre SOPRANO comme exemple avec notamment le système OpenAAL basé sur OSGi. Nos travaux portent l'accent non sur la santé mais sur le confort et la sécurité. Bien que les personnes les plus susceptibles de profiter de cette technologie soient des personnes âgées ou handicapées, notre intérêt est de développer une technologie *design for all* et basée sur des standards de l'industrie (KNX). L'environnement que nous souhaitons mettre en place est un habitat standard et non un habitat médicalisé comme dans *Aging In Place*.

La prise en compte de l'information audio est quasiment absente de tous les projets à l'exception notable de CompanionAble, de HERMES (Jianmin Jiang et Zhang, 2009) (pour la parole uniquement) et de (Brdiczka et coll., 2009). Pourtant ce canal apporte une information qu'il est parfois impossible à obtenir au moyen d'un autre capteur (p. ex. : cri) et qui peut présenter en outre une forte valeur sémantique (p. ex. : ordre domotique). Ceci confirme donc l'originalité du projet au niveau national et international.

Les capteurs employés dans les projets varient énormément en nombre et en type, cepen-

dant, on pourra remarquer l'omniprésence des capteurs de présence, des contacts de porte et compteurs énergétiques (eau, électricité, chauffage), en particulier dans les projets GER'HOME, Aging In Place, House_n, CASAS et iSpace. Ces capteurs sont aussi inclus dans notre projet. Cependant, l'utilisation de caméra est toujours sujette à débat et l'on pourra remarquer que même les projets les utilisant de manière continue comme House_n, GER'HOME et Aging In Place admettent que ces caméras ne sont pas toujours bien acceptées (cf. section 2.1.4.5). Par ailleurs, les dispositifs embarqués sur la personne constituent un autre type de capteurs intrusifs qui doivent être évités autant que possible (cf. section 2.1.4.5). Dans SWEET-HOME, aucun capteur intrusif n'est considéré comme entrée du système.

Les systèmes de décision présentés dans les projets sont fondés sur des modèles probabilistes ou des règles expertes (ou floues). Cette connaissance est souvent extraite par des méthodes de fouille de données (CASAS, House_n, Aging In Place). Pourtant, un grand nombre de niveaux d'abstraction doivent être considérés dans un environnement aussi riche et bruité qu'un habitat. Ces différents niveaux nécessitent l'emploi de méthodes adaptées et différentes comme le montre la variété des techniques employées. Nous discuterons toutes ces techniques dans le chapitre 4.

2.2 Analyse de contexte

2.2.1 Définition du contexte dans le cadre de l'Intelligence ambiante

McCarthy (1993) présente le *contexte* comme un ensemble d'entités mathématiques abstraites présentant des propriétés utiles pour les applications logiques de l'intelligence artificielle. Cependant, le terme contexte est utilisé en informatique avec une signification qui peut varier selon le domaine d'application. Par exemple, en Traitement Automatique du Langage Naturel (TALN), la notion de contexte est différente de celle utilisée dans le domaine des interfaces homme-machine. De plus, il convient de noter que non seulement il existe une définition propre à chaque domaine, mais qu'il est aussi possible de trouver plus d'une définition du contexte pour un même domaine. Néanmoins, nous pouvons dire que, dans tous les cas, le contexte est associé à l'interprétation d'une entité et à sa signification ; par exemple en ce qui concerne l'interprétation d'un mot en TALN, un mot peut avoir plusieurs significations, et son interprétation dépendre du contexte (donné par la phrase) dans lequel il se trouve.

Dans le cadre de l'intelligence ambiante, le contexte a été limité initialement à des informations spécifiques telles que la localisation (Schilit et coll., 1994; Brown et coll., 1997). Ensuite, d'autres éléments ont été ajoutés comme l'orientation, l'état émotionnel de l'utilisateur et la date (Dey, 1998). Par contre, simplement définir le contexte à partir d'une liste d'informations est trop restrictif car il y a des circonstances où les éléments en jeux pourraient ne pas appartenir à cette définition.

La définition du concept qui nous parait la plus pertinente dans l'état de l'art est celle donnée par Dey (2001) : *Le contexte est n'importe quelle information qui peut être utilisée*

pour caractériser la situation d'une entité[10]. Une entité peut être une personne, un lieu ou un objet considéré comme pertinent pour l'interaction entre l'utilisateur et l'application.

Nous trouvons cette définition plus générale et mieux adaptée à nos travaux de recherche. En effet, le contexte est toujours associé à une situation dans laquelle il est nécessaire de réduire l'ambiguïté, et le contexte n'est pas applicable que si une situation est susceptible d'avoir plusieurs interprétations. En outre, le contexte ne contient pas l'ensemble des informations disponibles pour le système, mais seulement un sous-ensemble qui est utile pour la désambiguïsation. Les informations composant le contexte ne sont pas les mêmes dans tous les cas, elles changent lorsque la situation évolue ; ainsi par exemple, le temps pourrait être l'élément le plus important du contexte dans une situation et être complètement insignifiant dans une autre.

Cependant, Abowd et coll. (1999) et Ryan et coll. (1997) ont retenu comme éléments les plus importants pour caractériser la situation d'une entité :

- le temps,
- la localisation,
- l'activité,
- et l'identité de la personne.

Ces éléments sont en relation avec les aspects essentiels du contexte décrits par Schilit et Theimer (1994) :

- « *Où es-tu ?* »,
- « *Avec qui es-tu ?* »,
- « *De quelles ressources disposes tu ?* ».

En outre, une application qui utilise le contexte doit connaître le *quand*, et le *quoi* (ce que la personne est en train de faire) pour déterminer le *pourquoi* une situation a eu lieu. Dans ce travail de recherche, nous utiliserons les mêmes éléments pour composer le contexte, sauf l'identité de la personne qui n'est pas importante puisque nous assumons la présence d'une seule personne dans l'environnement. Cependant, l'importance de chaque élément de contexte dépendra de la situation à évaluer.

2.2.2 Sensibilité au contexte

Les systèmes sensibles au contexte ou *Context Aware Systems* adaptent leur comportement à la fois selon plusieurs facteurs qui conditionnent leur environnement à un instant donné, selon les utilisateurs, leur localisation, les dispositifs accessibles, mais aussi selon les changements dans le temps que subissent ces facteurs (Schilit et coll., 1994). L'ensemble de ces éléments constitue le contexte. Pour être sensible au contexte, un système doit donc être capable d'obtenir ces informations, de les analyser, et de réagir de la façon la plus convenable pour l'utilisateur.

Selon Dey (2001), un système est sensible au contexte s'il utilise le contexte pour fournir des informations ou des services pertinents pour l'utilisateur, la pertinence dépendant de la

10. "Context is any information that can be used to characterize the situation of an entity"

tache de l'utilisateur.

La sensibilité au contexte concerne l'acquisition de contexte (par exemple au travers de capteurs), l'abstraction et la compréhension du contexte (en faisant le lien entre des événements détectés et des éléments du contexte), et l'adaptation de la réponse du système basée sur le contexte reconnu.

Un des problèmes le plus importants pour la recherche en systèmes sensibles au contexte est l'aspect spatio-temporel des informations qui composent le contexte. Par exemple, une consommation électrique importante, que l'on considèrerait normale à midi avec la présence de l'habitant, deviendrait *a priori* anormale la nuit alors que l'habitant est dans sont lit. Dans cet exemple, la temporalité du contexte peut être analysée à plusieurs échelles, si l'évènement a lieu de jour ou nuit (quand la personne est censée dormir), s'il a lieu en été ou en hiver (dans ce cas le le chauffage pourrait rester en route la nuit). Une modèle de contexte complet peut aussi considérer l'information relative aux événements qui ont amené à une telle situation, si cette consommation est restée constante sur un long intervalle de temps ou s'il s'agit d'une variation brusque.

Les systèmes sensibles au contexte peuvent être classifiés en trois groupes selon l'utilisation du contexte selon Schiele et coll. (2010) :

- **Présentation sensible au contexte :** l'application change d'apparence selon l'information de contexte. Il s'agit par exemple des systèmes de navigation qui changent le niveau de détail en fonction de la vitesse ou des applications multimodales qui peuvent changer leur sortie, notamment passage de l'audio à la vidéo, en fonction des bruits environnementaux.
- **Exécution automatique de services :** les applications sont déclenchées en fonction du contexte. C'est le cas des services de rappel qui notifient leurs utilisateurs différemment selon la localisation de ceux-ci, par exemple quand ils se trouvent dans un magasin, ou une salle de classe. La plupart des applications pour les maison intelligents appartiennent à cette catégorie, de même que les services qui détectent des circonstances où il y a un risque pour la sécurité ou le confort de l'habitant, et se servent du contexte (l'activité ou l'heure) pour mesurer le risque.
- **Marquage d'information de contexte pour une utilisation ultérieure :** ces applications permettent d'enrichir l'interaction des utilisateurs avec leurs environnements au travers d'objets virtuels. Des messages associés à certaines positions et à certains objets environnant l'utilisateur peuvent alors apparaître sur les interfaces dans certaines conditions.

2.2.3 Travaux sur la reconnaissance des situations en modélisation du contexte

La définition du terme *situation* a été abordée dans plusieurs travaux de l'état de l'art de l'intelligence artificielle et elle varie en fonction du domaine d'application, Singh et coll. (2012) ont fait une revue des ces définitions. La définition que nous avons trouvée la plus per-

tinente dans le cadre de nos travaux sur l'intelligence ambiante est celle de (Häussermann et coll., 2010) : *Une situation est la caractérisation d'une circonstance spécifique et récurrente dans le monde réel qui peut premièrement être décrite d'une façon idéale et qui deuxièmement sert de base d'évaluation pour l'adaptation et la réaction des applications sensibles au contexte* . Les auteurs considèrent la situation comme une description de conditions structurées (comportant les paramètres considérés importants) qui composent un modèle. Le remplissage du modèle avec des données concrètes et son traitement permettent de diagnostiquer l'existence d'une situation particulière, une instance de situation. De la même façon, nous estimons approprié de décrire une situation par un ensemble de conditions sur des événements qui modifient les valeurs des sources d'information mais aussi sur la relation temporelle ces événements. Ces conditions peuvent être interprétées comme des circonstances de risque dans lesquelles le système doit intervenir. Par conséquence, la reconnaissance de situations consiste à déterminer l'occurrence d'une instance de situation qui peut déclencher la prise de décision et l'analyse du contexte pour choisir la meilleure action à entreprendre.

Les travaux sur l'implémentation des systèmes sensibles au contexte dans des environnements perceptifs varient énormément selon les applications. Par contre, ils présentent plusieurs caractéristiques communes, comme les informations contextuelles qui sont les plus pertinentes pour analyser la situation courante. Dans la plupart des cas, le contexte est déterminé par la localisation, l'activité en cours, la date, et l'heure, comme nous l'avons décrit dans la section 2.2.1. Les premiers travaux se sont basés sur des représentations orientés vers l'application pour modéliser le contexte, ainsi la modélisation est intégrée aux modèles qui servent à faire l'inférence (Chen et Kotz, 2000). Le problème de cette approche est le manque de formalisation et d'expressivité (la diversité des concepts qui peuvent être représentées et communiquées grâce au modèle), en plus la difficulté de réutilisation du modèle dans une autre application. Par exemple, Schmidt et coll. (1999) ont proposé une application pour des dispositifs PDA qui change la taille de police selon l'activité de l'utilisateur et les conditions environnementales. Dans cette approche, les informations utilisées proviennent d'accéléromètres, de capteurs infrarouges et de microphones. L'inférence est exécutée par des cartes auto-organisatrices de Kohonen lesquelles utilisent les données brutes directement sans employer un modèle de représentation. D'autres systèmes suivent des démarches similaires pour déterminer le contexte (Yan et Selker, 2000; Dey et coll., 1999).

Puis les approches ont évolué pour séparer la représentation du contexte des modèles d'inférence, permettant ainsi d'avoir des représentations plus lisibles et faciles à modifier. Ainsi, un modèle entité-relation (ER) (Chen, 1976) a été présenté par Harter et coll. (1999) pour représenter les concepts liés au contexte dans le cas d'un système de suivi de l'utilisateur dans un bâtiment. La mise en œuvre de ces modèles permet de formaliser les relations entre les informations qui composent le contexte et de stocker les données reçues des capteurs d'une façon plus structurée afin de mieux les exploiter. D'autres travaux ont suivi une approche de modélisation orientée objet. Henricksen et coll. (2002) proposent une représentation basée sur UML spécifique aux systèmes sensibles au contexte, ce travail est en plus un

des premiers à prendre en compte la modélisation des imperfections caractérisant l'information du contexte dans les applications réelles. L'avantage des modèles orientés objet est qu'ils sont très expressifs et qu'ils offrent un cadre pour structurer les données acquises de l'environnement perceptif ; par ailleurs, les objets définis peuvent servir à la conception des logiciels implémentant un système sensible au contexte. Pourtant, dans ces modèles la réutilisation des connaissances est assez limitée et comme ce sont uniquement des modèles de représentation et de stockage, ils n'offrent pas de mécanisme de raisonnement.

Actuellement, la plupart des systèmes sensibles au contexte utilisent des approches logiques pour représenter les situations et le contexte. Les représentations évoluent alors de modèles orientés informations vers des modèles orientés connaissances. En plus de modéliser des relations complexes (par exemple temporelles ou hiérarchiques), il est possible de compter sur des mécanismes d'inférence pour obtenir des nouvelles connaissances à partir d'axiomes et de faits spécifiés dans les modèles. Parmi ces approches, le plus répandue est la modélisation par des ontologies, surtout quand la formalisation de la représentation est plus importante que l'inférence de nouvelles connaissances. Il est possible de classifier ces travaux en deux groupes. Le premier groupe correspond aux études qui effectuent l'inférence en utilisant des raisonneurs sémantiques ou la programmation logique, le deuxième groupe à celles qui utilisent les ontologies seulement comme moyen de représentation et comptent sur d'autres méthodes pour traiter les imperfections, telles que l'incertitude ou l'incomplétude, des informations dans le processus d'inférence. Dans le premier groupe, Liao et Tu (2007) présentent le système TempCRM qui modélise des événements temporels en utilisant des graphes RDF (*Resource Description Framework*) et comptent sur une ontologie pour représenter les concepts associés à une maison intelligente. L'inférence dans ce système se fait avec des règles logiques. Alors, même quand les auteurs ont défini des distributions probabilistes pour évaluer le dégrée de dangerosité d'une situation, l'incertitude n'est prise en compte qu'à la fin de la reconnaissance de situations sans considérer l'incertitude des informations d'entrée. Chen et coll. (2004) ont aussi employé des représentations ontologiques pour mettre en œuvre un système d'assistance pour les orateurs et participants d'une présentation. Cette fois, les auteurs comptent sur des règles logiques intégrées à l'ontologie grâce à l'outil Jess qui peut inférer des connaissance en utilisant les concepts et relations définis. ICONAWA (Yilmaz et Erdur, 2012) est un système qui propose des services et informations aux utilisateurs de dispositifs mobiles selon le contexte. Il a été conçu selon le même principe, avec une représentation ontologique des concepts et relations et l'utilisation des Jena, un moteur de règles, pour le raisonnement. En plus des ontologies, d'autres solutions logiques sont aussi importantes à considérer. Event Calculus (Kowalski et Sergot, 1986) est une approche qui formalise la définition d'événements complexes à partir de relations (temporelles ou spatiales) entre des événements atomiques. Ces définitions établies par des règles logiques permettent de définir les situations comme des événements complexes qui peuvent être reconnus en analysant la suite des événements provenant des capteurs (Chen et coll., 2008). Dans un autre travail, Mileo et coll. (2010) appliquent *Answer Set Programming* (ASP) pour l'implémentation de SINDI, un système pour la prédiction de situations de

risque dans une maison intelligente par la reconnaissance et l'évaluation de situations en contexte.

Les approches logiques, et plus spécialement les ontologies, offrent un cadre bien adapté pour représenter des connaissances relatives aux environnements perceptifs. Cependant, les mécanismes d'inférence qu'elles fournissent ne traitent pas des caractéristiques propres aux informations des applications réelles : imprécision, incertitude et incomplétude. Par contre, plusieurs travaux appliquent des méthodes d'inférence statistiques ou probabilistes indépendantes de l'ontologie mise en place pour la représentation. Ainsi, Gu et coll. (2004) ont implémenté le raisonnement en contexte par un réseau bayésien qui obtient les relations et les concepts stockés dans une ontologie. Par la suite, Gu et coll. (2005) ont présenté un cadre de travail qui modélise les éléments d'une maison intelligente par des ontologies et permet d'ajouter des modules d'inférence avec des méthodes personnalisées. Des règles de logique floue, qui exploitent la connaissance décrite par une ontologie, ont été mises en place pour traiter l'imprécision dans la reconnaissance de situations par Anagnostopoulos et coll. (2007).

2.2.4 Éléments du contexte

2.2.4.1 Localisation de l'habitant

Les techniques de localisation de personnes dans un environnement perceptif peuvent être divisées en deux catégories : celles qui utilisent des capteurs explicitement dédiés à cette tâche et portés par les personnes tels qu'un bracelet GPS et, celles qui utilisent des capteurs fournissant des informations implicitement porteuses d'information sur la présence d'une personne dans un espace restreint tels que des capteurs de présence infrarouges ou des caméras vidéos de surveillance. Les capteurs placés sur la personne sont souvent adoptés dans les situations où la personne a une activité sociale (visite de musée), pour des raisons professionnelles (nucléaire) de santé ou de sécurité (fugue de personnes âgées). Les techniques de localisation qui reposent sur ces capteurs obtiennent généralement de très bonnes performances en localisation (et identification), cependant elles ne sont pas adaptées à un usage informel et confortable. En effet, ces capteurs peuvent être lourds et encombrants et même gênants pour la personne, ils doivent de plus faire l'objet d'une vérification systématique des piles (sauf pour les capteurs passifs). Par ailleurs, si l'objectif de la domotique est d'améliorer le confort de vie, la contrainte d'un capteur portable, souvent oublié par les personnes (d'autant plus si elles sont âgées), représente une intrusion forte dans la vie intime de la personne. C'est la raison pour laquelle nous nous concentrerons dans cet état de l'art sur les techniques utilisant des capteurs placés à demeure dans l'environnement (vidéo, son, capteur de mouvements, contact de portes, etc.).

La localisation par analyse vidéo est une technique employée couramment pour la surveillance de bâtiments, par exemple, les trames de plusieurs caméras sont utilisées pour capturer les mouvements de la personne dans le projet d'assistance à domicile GER'HOME (Zouba et coll., 2009). Des capteurs environnementaux (placés sur des ouvertures de portes,

sur des fauteuils, etc.) permettent de détecter des interactions avec l'environnement. La reconnaissance d'événements complexes (activités de la vie courante, chutes) est faite par un système d'inférence dont la connaissance est représentée par des modèles hiérarchiques d'événements (un événement complexe est un ensemble de sous événements avec des contraintes temporelles et atemporelles). L'analyse vidéo est une modalité très intéressante pour la domotique et est utilisée dans de nombreux projets liés à cette thématique : Place_Lab (MIT) (Intille, 2002), Aware Home (Georgia Tech) (Bian et coll., 2005) et GER'HOME (Zouba et coll., 2009). Cependant le traitement vidéo nécessite des ressources en calcul et en stockage très importants qui impliquent un coût important d'installation et de maintenance. De plus, notre expérience nous a appris qu'installer des caméras d'enregistrement dans un logement peut être souvent perçu comme une trop grande intrusion dans la vie quotidienne.

Une autre source de localisation peut être dérivée des appareils électroménagers et de surveillance. Par exemple, dans le projet AILISA (Le Bellego et coll., 2006), des capteurs passifs infrarouges prévu pour l'éclairage automatique sont utilisés pour évaluer la position et l'activité de la personne. Par ailleurs, l'utilisation de matériel électroménager et de rangement peut aussi informer sur la localisation d'une personne dans l'habitat (si il y a ouverture de l'armoire à vêtement alors la personne est *a priori* dans la chambre). Cette information est généralement obtenue par des capteurs fixés sur les portes du mobilier ou sur les appareils ménagers considérés. Par exemple, dans (Vacher et coll., 2010a), un ensemble de capteurs de ce type est utilisé conjointement avec l'analyse audio pour reconnaitre les activités de personnes dans un habitat intelligent pour la santé. L'information d'utilisation des appareils électroménagers pourrait aussi se faire à moindre coût par des nouvelles techniques d'analyse qui permettent de détecter les signatures de mise en route de ces appareils sur le réseau électrique de l'habitat (Berenguer et coll., 2008). Cependant, comme la plupart des appareils sont maintenant programmables (p.ex. four électrique) et/ou utilisable à distance (p.ex. téléphone), l'information de localisation devient donc moins sûre à moins d'utiliser un système domotique centralisé comme proposé dans le projet SWEET-HOME.

Une autre modalité intéressante en domotique est l'analyse du canal audio, qui, en plus de permettre une commande vocale, peut apporter diverses informations sonores, telles que la détection de bris de verre, de claquement de portes, etc. (Vacher et coll., 2010a). De part sa nature omnidirectionnelle, le microphone est aussi un capteur d'événements et de localisation prometteur. Par exemple, Bian et coll. (2005) ont installé une grille de microphones dans une pièce pour déterminer l'origine d'un son dans un espace à 3 dimensions avec une erreur moyenne de moins de 27 cm. Comme souligné par les auteurs, l'information audio nécessite beaucoup moins de bande passante que l'information vidéo et permet de détecter facilement certaines activités (conversations, sonneries de téléphone). Cependant, cette solution de grille de micros est trop complexe et trop couteuse à mettre en place dans un appartement classique. De plus, si la vidéo est sensible aux changements de luminosité, le canal audio est sensible au bruit environnemental. Dans (Vacher et coll., 2010b), des enregistrements audio impliquant une quinzaine de personnes ont été effectués pendant plus de 11 heures dans un appartement, ceci a permis la mise en évidence d'un nombre très important de bruits ex-

térieurs perturbant le système de reconnaissance. Le canal audio, bien qu'étant une modalité pertinente et peu couteuse, est donc une source bruitée et parfois peu informative sur la source du bruit.

Au fil de cet état de l'art, il apparait donc qu'aucune source ne permet à elle seule de réaliser une localisation de manière sûre et à peu de frais. Il est donc important de mettre en place un système de localisation multisource qui tirerait partie des redondances et des complémentarités des sources choisies. Plusieurs travaux ont avancé dans ce sens, en utilisant principalement des méthodes de classification probabilistes telles que les réseaux bayésiens (Dalal et coll., 2005) ou les modèles de Markov (Kröse et coll., 2008). Cependant, l'information temporelle est extrêmement pauvre et rarement prise en considération dans ces modèles. Des approches à base de règles ont été proposées (Zouba et coll., 2009; Duong et coll., 2009) mais l'incertitude sur les données n'est pas considérée. Dans (Hong et coll., 2009), la théorie de Dempster-Shafer est appliquée pour fusionner l'information provenant de plusieurs capteurs afin de combiner des preuves de manière hiérarchique pour reconnaitre deux activités culinaires. Cependant, l'information temporelle n'est, encore une fois, pas prise en considération.

Duong et coll. (2009) prennent en compte l'aspect temporel à travers d'un modèle semi-Markov caché hiérarchique qui fait la reconnaissance des activités à partir d'une séquence d'événements mais en modélisant aussi le temps passé dans chacun des états (chaque état représente une sous-activité) ; pourtant, dans notre cas, il sera difficile de prédire la localisation présente en évaluant une séquence de localisations passées parce que le nombre important de trajets possibles augmenterait l'incertitude.

2.2.4.2 Reconnaissance de l'activité

La reconnaissance automatique de l'activité humaine (manger, discuter, regarder la télévision, etc.) peut être définie comme l'identification d'une séquence d'actions atomiques (prendre un ustensile de cuisine, faire couler l'eau, s'allonger, etc.). Ceci implique d'abstraire les signaux bruts en symboles (propositions) étiquetés temporellement (p. ex. : claquement de porte à 11h32), les signatures de situations particulières d'événements atomiques sont détectées à travers un processus d'abstraction hiérarchique. Par exemple, les mouvements détectés dans la chambre à coucher peuvent faire partie de l'activité « se lever » qui elle même peut faire partie d'un plan de journée (p. ex. : matinée du dimanche). La reconnaissance automatique d'activité est l'un des secteurs de recherche les plus actifs et les plus ambitieux en raison de la quantité importante de bruit dans les données et de la difficulté de modélisation des situations. En effet, pour une même personne, une activité peut se dérouler de nombreuses façons.

La reconnaissance d'activité humaine à partir de capteurs est un domaine scientifique en pleine effervescence avec de nombreuses applications dans le domaine de la sécurité et de la vidéo-surveillance de lieux publics (Fusier et coll., 2007; Rota et Thonnat, 2000). Cette recherche s'est récemment étendue à l'assistance à domicile (Tapia et coll., 2004; Storf et coll., 2009; Zouba et coll., 2009), à l'assistance et à la surveillance dans les unités de soins

spécialisés (Parlak et coll., 2011), ainsi qu'à la surveillance et à l'évaluation de procédés dans l'industrie (Ward et coll., 2006). Dans le cadre de notre travail, la reconnaissance d'activité est importante pour déterminer le contexte dans lequel le système doit agir.

L'environnement impose des contraintes au niveau des capteurs ainsi qu'au niveau des techniques employées pour la reconnaissance. La plupart des progrès apportés dans le domaine de la reconnaissance d'activité, comme dans la localisation, proviennent de la vision par ordinateur (Aggarwal et Ryoo, 2011). Cependant, l'installation de caméras chez l'habitant soulève des questions éthiques, c'est pour cela que la solution que nous avons développée dans ce travail consiste à compléter les capteurs conventionnels de la domotique (détecteurs de présence infrarouges, interrupteurs, etc.) par des microphones pour permettre à l'utilisateur de maîtriser son environnement grâce à une reconnaissance vocale et un contrôleur intelligent. Le contrôleur intelligent doit prendre les décisions en fonction du contexte. Par exemple, l'allumage d'une lampe se fera de manière différente selon que l'utilisateur interrompt son sommeil ou qu'il est en train de manger, ce ne sera pas non plus la même lampe qui sera utilisée. La reconnaissance d'activité constitue donc un élément essentiel du système.

Un nombre très important de méthodes ont été proposées pour la reconnaissance d'activité à partir de capteurs non visuels. Ces méthodes peuvent être classifiées en trois catégories, statistiques, probabilistes ou logiques. Le premier groupe inclut des méthodes d'apprentissage automatique. Par exemple, Fleury et coll. (2010) proposent l'application des Séparateurs à Vaste Marge (SVM) pour classifier des Activité de Vie Quotidienne (AVQ) ou *Activities of Daily Living* (ADL) à partir des données obtenus d'un environnement perceptif réel. Les méthodes statistiques, comme SVM ou les réseaux de neurones, offrent une bonne performance, par contre leur représentation de l'incertitude est implicite ; de plus, leurs modèles manquent de lisibilité et sont difficiles à interpréter.

L'approche logique, celle du deuxième groupe, offre un cadre mieux adapté pour modéliser des connaissances explicites. Les ontologies ont été appliqués à ce problème car elles présentent l'avantage de la lisibilité et la définition formelle des connaissances, et sont bien adaptées à la classification d'une séquence d'évènements dans une activité par un outil de raisonnement (Chen et Nugent, 2009). La reconnaissance à base de règles logiques expertes est très répandue (Storf et coll., 2009; Zouba et coll., 2009). Les règles logiques permettent de définir des relations complexes entre les éléments concernés par une activité tels que les relations temporelles entre les évènements (Augusto et Nugent, 2004). Cependant ces méthodes ne prennent pas en compte l'incertitude.

L'incertitude a été traité par des méthodes probabilistes comme par exemple les réseaux Bayésiens utilisés par van Kasteren et Krose (2007). Une approche dynamique, qui considère l'aspect temporel, consiste à modéliser les activités par un modèle de Markov caché ou *Hidden Markov Model* (HMM) (Naeem et Bigham, 2008; Duong et coll., 2009). Plusieurs variantes permettent plus de souplesse dans l'ordre des séquences (par exemple les HMM hiérarchiques (Naeem et Bigham, 2008)) ou la modélisation de la durée des activités Duong et coll. (2009). Cependant, malgré la prise en compte de l'incertitude, ces modèles nécessi-

tent beaucoup de données d'apprentissage, données qui sont très rares dans le domaine de l'intelligence ambiante. De plus, leur lisibilité est aussi pauvre et il est difficile de définir des relations complexes entre les variables du problème. D'autre part, il est difficile de prendre en compte de la connaissance *a priori* dans ces modèles.

Une nouvelle méthode utilisant les réseaux logiques de Markov ou *Markov Logic Network* (MLN) a été appliquée avec succès à ce domaine (Tran et Davis, 2008) ; elle combine la puissance d'expressivité de la logique du premier ordre à la capacité à traiter l'information incertaine des méthodes probabilistes. Ceci présente de multiples avantages : le modèle est défini dans un langage logique formel qui le rend facilement vérifiable et réutilisable (Artikis et coll., pear), l'incertitude est prise en compte de manière systématique, et enfin l'apprentissage du modèle permet l'inclusion aisée de connaissances *a priori*.

2.2.5 Reconnaissance de la parole et des sons comme support à la reconnaissance du contexte

Parmi les informations permettant d'effectuer la reconnaissance de contexte, les événements captés par les microphones ont une importance particulière dans la solutions proposée dans ce travail. Les événements audio fournissent des informations très riches pour la désambiguïsation qui constitue le but ultime de l'analyse du contexte lors de l'évaluation d'une situation. Dans l'intelligence ambiante, l'analyse audio peut être scindée entre reconnaissance de la parole et reconnaissance des sons. Ces deux domaines couvrent naturellement des aspects différents concernant les maisons intelligentes ; en même temps, ils partagent plusieurs défis en commun. Ainsi, dans les conditions réelles, le traitement des canaux audio est toujours affecté par la présence des bruits environnementaux (téléviseur, radio, ustensiles électroniques variés), par l'acoustique de la pièce, la position de la source sonore (l'habitant ou l'appareil) et son orientation par rapport au microphone.

La Reconnaissance Automatique de la Parole (RAP) est un domaine d'étude très ancien et bien établi qui a déjà franchi plusieurs étapes. Les premiers systèmes (1952-1968) utilisaient des approches heuristiques (bancs de filtres et circuits logiques), ils permettaient simplement de reconnaître les 10 chiffres. La deuxième génération (1968-1980) était fondée sur des méthodes de reconnaissance de formes dans le domaine spectral (DWT, FFT, LPC), elle était utilisée pour la reconnaissance de mots isolés. Les systèmes actuels utilisent des méthodes d'analyse statistique (chaînes de Markov cachées - HMM, n-gram, réseau neuronaux), ils peuvent reconnaître de la parole continue. C'est l'utilisation des méthodes probabilistes qui a permis ce grand progrès même si actuellement une stagnation des performances est constatée malgré l'introduction de nouveaux procédés comme les approches discriminantes, l'apprentissage robuste, l'adaptation au locuteur et la transcription riche. Ces développements ont donc conduit à des résultats significatifs et ont permis à la RAP d'être un composant à part entière de beaucoup de produits industriels. Diverses utilisations de la reconnaissance de parole dans les environnements perceptifs sont envisageables comme la commande vocale de dispositifs et le dialogue entre l'habitant et le système de contrôle domo-

tique. Les interfaces ayant pour base la voix pourraient être très importantes pour les personnes handicapées (ayant des problèmes de vision et mouvement) quand elles ont besoin de contrôler leur environnement, surtout dans les cas où l'utilisation des interfaces tactiles devient difficile à opérer. Lines et Hone (2006) illustrent cette application par l'implémentation d'un système interactif d'alarme pour faciliter l'indépendance des personnes âgées. Par exemple, quand l'habitant se réveille la nuit, il peut demander au système d'allumer la lumière au lieu de chercher un interrupteur dans le noir. Pour la reconnaissance de situations en contexte, la reconnaissance des mots de détresse peut être utilisée pour estimer l'état courant dans l'environnement perceptif. En plus de permettre la reconnaissance de situation, l'aspect émotionnel de la parole peut servir à désambiguïser le contexte. Par exemple, les expressions de joie ou tristesse lorsque l'habitant parle peuvent être considérées lors de la prise de décision. Parmi les travaux qui ont utilisé la reconnaissance de parole dans des environnements perceptifs, certains visent a améliorer l'analyse d'un aspect concret du contexte. Ainsi, dans le cadre de l'assistance à domicile Vacher et coll. (2010a) décrivent un système pour la reconnaissance d'activité humaine à partir des informations de son et parole reconnues par le système RAPHAEL (Vaufreydaz et coll., 2000) qui analyse le signal de parole et fournit une meilleure hypothèse pour identifier des mots clefs de détresse. La simulation de l'interaction humain vers humain par l'utilisation des systèmes multimodaux est une autre application de la reconnaissance de parole (López-Cózar et Callejas, 2010; Hamill et coll., 2009). Un de principaux défis de la reconnaissance de parole dans les environnements perceptifs est la reconnaissance en conditions distantes. Les applications usuelles des systèmes de RAP se placent dans le cas où le microphone est placé à quelques centimètres de la bouche du locuteur. Lorsque les microphones sont placés au plafond et dirigés vers le sol, il n'y a plus un trajet unique de la source vers le récepteur mais une combinaison de trajets qui font intervenir l'acoustique de la salle. Par ailleurs le cône d'émission de la parole est très fermé pour les hautes fréquences, la parole recueillie comportera donc plus ou moins de hautes fréquences suivant la position de l'axe d'émission du locuteur par rapport à celui du microphone.

La classification des sons de la vie quotidienne dans les maison intelligentes n'est pas utile pour contrôler directement l'environnement mais elle peut servir à la reconnaissance de situations et la désambiguïsation du contexte. La surveillance par l'analyse de sons pour détecter des situations de détresse a été proposé par Istrate et coll. (2006) et Rougui et coll. (2009). D'ailleurs, il existe d'autres travaux qui utilisent des microphones et accéléromètres pour détecter des chutes (Litvak et coll., 2008; Popescu et coll., 2008). La reconnaissance des éléments du contexte peut être améliorée par l'inclusion du son, la localisation de l'habitant par exemple (Bian et coll., 2005). Son application à la détection automatique des activités a aussi été testée par Chen et coll. (2005) et Vacher et coll. (2010a), dans ce cas l'information des événements sonores est souvent complétée par les informations provenant du réseau domotique. La reconnaissance des sons est relativement neuve comme problème de recherche, cependant la détection et l'analyse de scènes acoustiques (Niessen et coll., 2008) est l'objet d'un intérêt récent qui se traduit par l'organisation d'une compétition or-

ganisée par l'*IEEE ASP Challenge* au printemps 2013 (Giannoulis et coll., 2012). La plupart des méthodes appliquées dépendent de l'apprentissage des modèles probabilistes à partir d'un corpus annoté. Pourtant, des expérimentations ont montré que la grande diversité de classes de son permet difficilement d'apprendre des modèles *a priori* pour les utiliser dans un milieu réel (Vacher et coll., 2011b). De plus, la durée de chaque son dans les corpus est très courte et le nombre d'instances est souvent insuffisant pour apprendre des modèles statistiques fiables. Ces problèmes compliquent l'utilisation de méthodes classiques d'apprentissage automatique.

2.3 Prise de décision dans les environnements perceptifs

Dans cette section, nous allons aborder le problème de la prise de décision en contexte dans les environnements perceptifs. Les systèmes de prise de décision traditionnels sont conçus pour aider les personnes par l'automatisation du processus d'inférence de la meilleure action à entreprendre. Pourtant, dans les maisons intelligentes, pour exécuter l'action correcte, les systèmes de contrôle doivent prendre en considération le contexte et agir de façon proactive. Nous allons commencer par décrire les systèmes de prise de décision en général, et ensuite nous donnerons une description des particularités de la prise de décision dans les environnements perceptifs avant de décrire des travaux existants.

2.3.1 Définitions

La prise de décision peut être considérée comme le processus qui va permettre de choisir une action à accomplir parmi plusieurs alternatives. Holtzman (1989) remarque que "la prise de décision signifie concevoir et s'engager à une stratégie d'allocation irrévocable de ressources précieuses" [11]. En fait, dans de nombreux cas, les actions résultant de la décision ne peuvent être réalisées sans entraîner un coût (le prix pour l'utilisation d'une ressource). Nous allons maintenant définir les éléments les plus basiques qui interviennent dans la prise de décision.

Définition 1 (Décideur) *Le décideur est un agent (ou un ensemble d'agents) qui prend (ou prennent) une décision en fonction d'un problème donné. Cet agent peut être une personne ou un logiciel. Il existe d'autres agents concernés par la décision prise et chacun a un rôle spécifique, par exemple celui qui peut se bénéficier du résultat de la décision, les experts du domaine, ou les ingénieurs travaillant dans ce domaine.*

Définition 2 (Alternatives) *Les alternatives sont les actions possibles que le décideur peut exécuter. Quand une alternative est choisie, une sortie doit être générée.*

Définition 3 (Sorties) *Les sorties du système de décision n'ont pas le même dégré de désirabilité. Quand il n'y a pas d'incertitude, chaque alternative correspond à une sortie associée, et le*

11. "Making a decision means designing and committing to a strategy to irrevocably allocate valuable resources"

choix peut être fait à travers une mesure de désirabilité pour évaluer chaque sortie. Dans des circonstances où la connaissance est affectée par l'incertitude, il n'existe pas une correspondance entre une alternative et une sortie car cette relation n'est pas déterministe.

Définition 4 (Situation de décision) *La situation de décision est constituée par l'état de l'ensemble des variables à l'instant courant, elle doit être identifiée pour chaque problème de décision. Ainsi, chaque problème de décision est spécifique à une situation particulière.*

Les systèmes formels de prise de décision sont souvent implémentés en utilisant des variables et domaines de valeur pour chaque élément décrits ci-dessus.

2.3.2 Défis des systèmes de prise de décision

L'implémentation de systèmes de prise de décision nécessite de relever des défis, nous allons maintenant passer en revue les principaux de ces défis.

Défi 1 (Incertitude) L'incertitude est une des caractéristiques les plus communes qui affecte les systèmes dynamiques. L'incertitude est tout d'abord produite par l'imperfection des informations, mais pas seulement. Elle est aussi introduite par l'insuffisance de la connaissance concernant la corrélation existante entre les alternatives et les sorties.

Défi 2 (Adaptabilité) Un système de prise de décision doit être capable de s'adapter aux changements opérés dans la connaissance, dans les flux d'information, ainsi que dans le cadre opérationnel (Druzdzel et Leong, 2005). Le système doit être suffisamment souple pour permettre des modifications dans les processus de décision.

Défi 3 (Gestion des connaissances) La prise de décision est une tâche qui demande l'utilisation de connaissances particulières : l'identification du problème, l'acquisition des informations et des évidences, la prédiction… À cause de l'hétérogénéité des sources d'information, l'efficacité de la gestion de connaissance peut être assurée seulement par l'utilisation d'un langage commun et formel. La difficulté est alors de trouver un équilibre entre l'expressivité et la souplesse parce que plus un langage est expressif, moins il est malléable.

Défi 4 (Collaboration) Pour être efficace, un système de prise de décision doit offrir une plate-forme permettant la collaboration entre les différents agents qui participent au processus de décision. Le décideur peut avoir besoin de coordonner, négocier, ou synchroniser ses activités avec les autres agents.

Défi 5 (Intelligence) L'intelligence inclut les capacités du système pour raisonner et faire des inférences au travers d'algorithmes plus ou moins sophistiqués.

Défi 6 (Transparence) La transparence est la capacité du système à permettre aux agents de comprendre les mécanismes du processus de décision. En plus, il est souhaitable d'avoir une explication claire des actions prises par le système.

2.3.2.1 Prise de décision en contexte

D'après Simon et Dillon (1960), il y a trois phases dans la prise de décision : l'intelligence (la compréhension de la situation), la conception, et le choix. Dans la première phase, le décideur identifie une situation laquelle est une divergence entre l'état courant et un état désiré. À partir de l'analyse de la situation, des alternatives pour éliminer la divergence sont générées et un modèle de décision est créé dans la phase de conception. Finalement, lors du choix une décision, est prise parmi toutes les alternatives.

En comparaison avec ce processus de décision typique, la prise de décision en contexte incorpore des éléments qui permettent de mieux caractériser la situation courante :

- premièrement par la reconnaissance automatique du contexte des agents directement impliqués dans le processus lors de la phase d'intelligence, dans le cas des environnement perceptifs cet agent est l'utilisateur ;
- deuxièmement dans la phase de conception, par la prise en compte des informations inférées qui sont inclues lors de la construction du modèle de décision.

Une caractéristique importante de ce processus est la grande sensibilité au contexte : il est nécessaire de détecter automatiquement des informations de contexte telle la localisation, le temps ou l'identité, de plus cela doit être fait de façon transparente pour l'utilisateur. Les paramètres obtenus à partir des information de contexte servent à déterminer le modèle de décision qui s'adapte le mieux à la situation courante. Même les degrés de préférence des sorties du système sont évalués en utilisant l'information contextuelle. La figure 2.5 montre un schéma de prise de décision en contexte.

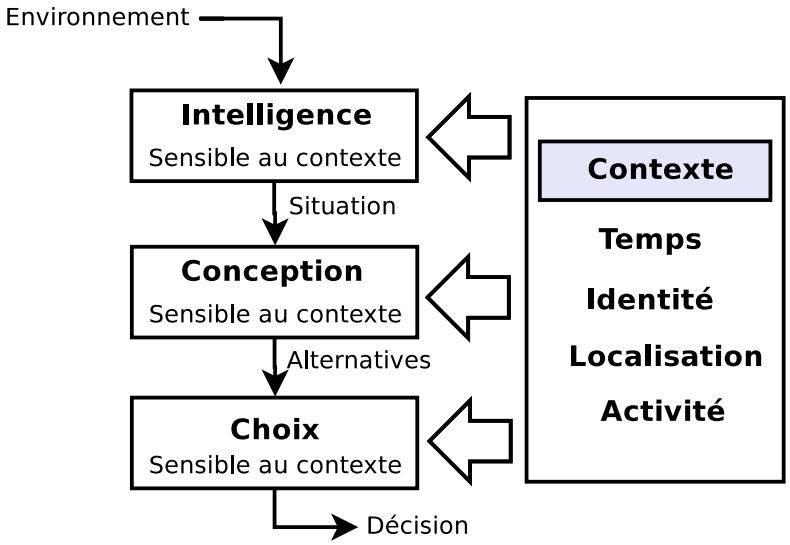


FIGURE 2.5: Prise de décision en contexte

2.3.3 Travaux existants sur la prise de décision

Concernant l'implémentation de systèmes de prise de décision, nous pouvons observer que la plupart des travaux se centrent soit sur la représentation formelle et les modèles ex-

pressifs, soit sur des modèles permettant de traiter l'incertitude propre aux systèmes dynamiques.

2.3.3.1 Utilisation de systèmes fondés sur des règles logiques

Le premier de ces deux groupes concerne des systèmes qui implémentent des règles logiques fondés sur les systèmes d'aide à la décision classiques. Un système expert qui suit une approche relationnelle utilisant des règles de la logique floue mais adaptée pour la décision multi-objectifs a été proposée par Rasmy et coll. (2002). Dans une autre étude, Moore et coll. (2011) présentent un étude sur l'application de règles logiques et la logique floue pour modéliser les relations conditionnels entre les situations et les actions à effectuer lors de la prise de décision. Il s'agit dans ce cas des règles ECA (Événement-Condition-Action) qui vont être déclenchées lors de l'arrivée d'événements spécifiques, ensuite la condition est évaluée en considérant le contexte, et finalement une action est exécutée comme résultat du processus de décision. En fait, dans ce type d'implémentation la décision est prise lors de l'évaluation de la condition, la sémantique des préférences selon la situation courante est spécifiée dans la condition.

2.3.3.2 Utilisation de systèmes fondés sur des ontologies

La représentation des connaissances par ontologies a été largement appliquée dans le domaine de la prise de décision. Par exemple, UbiDSS (Kwon et coll., 2005) est un système expert qui aide les utilisateurs de dispositifs mobiles à mieux effectuer leurs activités selon le contexte dans lequel ils se trouvent. À cette fin, le système se connecte à une base de connaissances implémentée par une ontologie qui garde les préférences de l'utilisateur et la description des éléments contextuels. L'outil CAMPUS (Wei et Chan, 2010) utilise aussi des ontologies pour capturer les sémantiques sous-jacentes aux entités d'une application mobile, puis quand les situations sont reconnues, les décisions à prendre sont adaptées en temps réel conformément au contexte de l'utilisateur. Dans le cadre des environnements perceptifs, Khattak et coll. (2010) ont présenté un système sensible au contexte qui considère principalement l'activité réalisée par l'habitant. L'information nécessaire pour inférer l'activité et les activités reconnues sont stockées dans l'ontologie et traitées à travers SPARQL, tandis que la décision est implémentée en règles de logique descriptive qui utilisent directement l'information de l'ontologie. Une approche qui permet de créer des règles plus complexes dans les ontologies est SWRL, cette méthode est utilisée par Ruan et coll. (2010) dans une application pour la prise de décision quand les préférences des utilisateurs d'un système sont en conflit. Dans tous les travaux qui suivent une approche logique, les imperfections des informations ne sont pas prises en compte pour traiter l'imprécision des termes liée à l'analyse de contexte et la prise de décision, sauf en cas d'utilisation de la logique floue.

2.3.3.3 Utilisation de systèmes fondés sur des méthodes statistiques ou probabilistes

Il y a plusieurs études dans le domaine de la prise de décision qui suivent un approche probabiliste pour traiter l'incertitude inhérente au comportement dynamique des systèmes sensibles au contexte. Ainsi Lee et Cho (2012) ont montré l'intérêt des réseaux bayésiens dans la prise de décision en contexte à travers de la modélisation des relations causales, tandis que Lacey et MacNamara (2000) ont implémenté un système, basé sur les réseaux bayésiens, qui utilise des informations de contexte et prend des décisions pour assister des personnes âgées. Les Diagrammes d'Influence (DI) sont des modèles adaptés à la décision qui résultent d'une extension des réseaux bayésiens. Ils ont été proposés comme solution pour la prise de décision sous incertitude dans des systèmes sensibles au contexte. Par exemple, Mitra et coll. (2011) utilisent l'approche des *Context spaces* pour modéliser les informations concernant le contexte, tandis que la décision est traité par une méthode de DI. Une autre solution, aussi dans le domaine des environnements perceptifs, utilisant des DI a été développée par Carolis et Cozzolongo (2004). Il s'agit d'un système multi-agents pour assurer le confort, la sécurité et le bien-être de l'habitant. Nishiyama et coll. (2011) ont proposé aussi les DI pour assurer le confort de l'habitant en régulant le système de chauffage pour avoir la température la plus convenable selon le contexte. Ce dernier travail exploite une caractéristique importante des DI, la possibilité d'inclure le risque comme facteur dans le processus de décision. Ici le risque est de dépenser trop d'énergie électrique lors de réglage de la température.

2.3.3.4 Conséquences pour un choix optimal

Comme il a été exposé dans la section 2.3.2, idéalement la prise de décision doit considérer non seulement le traitement de l'incertitude, mais aussi la représentation formelle des connaissances. Quelques travaux ont fait le point sur l'intersection de ces deux problèmes. Yang et Calmet (2006) ont proposé un outil pour les systèmes d'aide à la décision qui permet de construire des DI à partir des connaissances exploitées dans une ontologie. Un approche similaire a été employée par Cornalba et coll. (2008) pour un système d'aide à la décision pour la gestion de risque lors de traitements par hémodialyse. La combinaison de la représentation par ontologies et des méthodes probabilistes qui génèrent des modèles à partir des connaissances stockées est très intéressant pour traiter les deux problèmes fondamentaux que nous avons décrits. Par contre, quand des modèles probabilistes sont crées, ils ne sont pas capables de prendre en compte toutes les relations complexes possibles qui peuvent être spécifiées dans un modèle logique. Ainsi, par exemple, les réseaux bayésiens peuvent seulement inclure les relations causales spécifiées dans le modèle logique.

2.4 Synthèse et conséquences

Au cours de ce chapitre sur l'état de l'art, nous avons exposé les définitions essentielles permettant de comprendre le domaine dans lequel nos travaux sont réalisés, l'intelligence

ambiante. L'intelligence ambiante doit son développement à l'évolution des dispositifs électroniques, des réseaux de communication, mais aussi des méthodes de l'intelligence artificielle permettant de mieux utiliser les informations de l'environnement afin d'améliorer la qualité des services rendus aux utilisateurs. En général, l'implémentation de systèmes d'intelligence ambiante pose des défis importants tels que le raisonnement basé sur des informations incertaines et le traitement d'événements temporels. Les environnements où cette discipline peut être mise en œuvre sont très variés : maisons, bureaux, salles de classe, etc. Nous avons fait une description d'environnements perceptifs particuliers, en nous restreignant à celui des maisons intelligentes.

Une maison intelligente est un appartement équipé de systèmes informatiques pour faciliter la vie de ses habitants. Les buts d'une maison intelligente sont principalement d'assurer la sécurité et le confort. Pour cela, diverses taches peuvent être mises en place : la surveillance pour les personnes avec perte d'autonomie, la mise en communication avec l'extérieur, l'assistance dans les activités de vie quotidienne, parmi d'autres. Les projets de maison intelligente qui ont été développés sont nombreux, nous avons fait une description sommaire de ceux qui nous paraissaient les plus pertinents pour notre recherche. Parmi les projets qui nous avons décrits, il est possible remarquer que les solutions plus génériques et complètes sont proposées par les projets qui ne sont pas orientés exclusivement vers une application médicale. Les capteurs employés dans les projets varient énormément en nombre et en type. Bien que l'utilisation de cameras ne soit pas toujours acceptée, plusieurs projets l'utilisent de manière continue (House n, GER'HOME, Aging In Place). Par ailleurs, les dispositifs embarqués sur la personne constituent un autre type de capteurs intrusifs, comme c'est le cas dans Aging In Place, ils n'ont donc pas été retenus dans nos travaux.

Le contexte, l'ensemble des informations qui servent à caractériser la situation d'une entité, est un concept très important pour la bonne réalisation de systèmes d'intelligence ambiante. En effet, les systèmes sensibles au contexte agissent sur l'environnement en analysant le contexte courant de l'utilisateur au lieu de simplement suivre des actions préalablement définies, leur action correspond donc mieux aux souhaits de l'utilisateur. Dernièrement, la représentation par ontologies devient la norme pour la modélisation du contexte parce qu'elle offre un cadre formel et expressif pour partager la connaissance. La plupart des travaux de la littérature sur les environnements perceptifs signalent que les informations les plus importantes à considérer pour modéliser le contexte sont : la localisation, le temps, l'identité, et l'activité courante. Nous avons aussi retenu ces éléments pour notre implémentation, sauf en ce qui concerne l'identité car nous considérons la présence d'une seule personne dans l' environnement à un instant donné. La localisation peut être estimée avec une grande exactitude si on considère l'utilisation de capteurs portés par la personne, mais leur acceptation étant très limitée, nous ne les avons pas retenus.

L'état de l'art présente des travaux consacrés à l'inférence de la localisation à partir des données issues de capteurs environnementaux, mais dans la plupart de ceux-ci, l'aspect temporel n'est pas pris en compte. La reconnaissance d'activité est souvent traitée comme une application du domaine de la reconnaissance à partir de données vidéo, dans les autres

cas, elle nécessite des méthodes permettant de fusionner des informations issus de sources hétérogènes souvent incertaines et affectées par le bruit.

Lors de l'interaction entre un système sensible au contexte et l'utilisateur, la prise de décision est fondamentale pour choisir l'action correcte à effectuer et rendre le service approprié. La décision est souvent confondue avec la classification, en conséquence nous avons défini les éléments qui caractérisent un problème de prise de décision et aussi les principaux défis qui sont étudiés dans les applications d'intelligence artificielle. En outre, la prise de décision en contexte est un processus dynamique car elle requiert des informations propres à la situation comme par exemple le degré d'incertitude de cette situation. Nous avons classifié les travaux de l'état de l'art en deux catégories selon les aspects qu'ils couvrent le mieux : la représentation de connaissance et l'inférence logique d'un coté, et de l'autre l'application des méthodes statistiques qui sont mieux adaptées pour les systèmes ayant des variables stochastiques. L'intégration des modèles logiques et statistiques est un sujet très étudié en intelligence artificielle depuis ces dernières années. Les systèmes sensibles au contexte ne sont pas étrangers à ces développements, et, nous trouvons dans la littérature quelques travaux qui appliquent des méthodes d'inférence statistique utilisant des modèles de représentation logiques et surtout des ontologies.

CHAPITRE 3

Contrôle intelligent de la domotique

Les méthodes développées dans ce travail de recherche visent à faciliter à chaque instant le pilotage de son environnement par l'habitant, quelque soit sa position dans le logement et en utilisant l'interface que la personne jugera la plus naturellement adaptée. Il est clair qu'une personne mal voyante sera naturellement plus encline à utiliser une interface audio et non une interface tactile. De même, dans le cas d'une personne qui se réveille en pleine nuit, il est naturel de penser qu'une commande d'éclairage de la pièce par la voix sera préférée à la recherche d'un interrupteur dans l'obscurité. La possibilité de piloter la domotique par différents modes d'interface adaptées à la personne et à la situation va avoir des conséquences sur le système de pilotage de la domotique. Ce dernier sera confronté à des situations complexes, il devra donc s'appuyer sur un contrôleur intelligent qui prendra la forme d'un composant de décision couplé à plusieurs modules d'inférences et capable d'interpréter les ordres de l'utilisateur. Malgré cette complexité, le contrôleur devra assurer une grande réactivité aux demandes de l'utilisateur. En complément de cet aspect « réponse au requêtes », le contrôleur devra aussi constamment évaluer la situation courante de façon à anticiper les risques entraînés par la situation. Par exemple, le système devra détecter que la personne sort de son appartement en ayant laissé en route les plaques électriques, ou alors détecter que la personne est en train de se coucher alors que la porte d'entrée n'est pas fermée à clé.

Pour atteindre ces objectifs, il est nécessaire de pouvoir représenter et raisonner sur un grand nombre de données et de connaissances, le plus souvent temporelles, à des niveaux d'abstraction différents. Par ailleurs, les décisions prises par le contrôleur doivent tenir compte de la situation courante dans laquelle un ordre est donné, ce type de situation nous place d'emblée dans le traitement de données sensible au contexte ou *context-aware computing* (Soylu et coll., 2009). Par exemple, lorsqu'un ordre vocal « allume la lumière » est reçu, le contrôleur doit tout d'abord non seulement déterminer quelle est la lampe qu'il faut allumer mais aussi fixer l'intensité qui correspond le mieux à la situation actuelle. Si la situation correspond au cas où la personne est en pleine activité dans la chambre, c'est le plafonnier qui devra être allumé à forte intensité ; par contre, si la personne se réveille au milieu de la nuit, il faudra privilégier l'éclairage de la lampe de chevet avec une faible puissance.

L'environnement domotique dans lequel le contrôleur a été mis en œuvre dans notre étude est défini et décrit dans la section 3.1. Ensuite, la section 3.2 présentera certains exemples type de scénarios correspondant à l'utilisation prévue. Nous décrirons les informations

inférées par le contrôleur et la façon dont elles sont structurées en section 3.3. Finalement, nous décrirons l'architecture fonctionnelle dans la section 3.4 en détaillant plus particulièrement l'architecture logicielle du contrôleur intelligent dans la sous-section 3.4.2. La dernière section sera consacrée à la représentation des connaissances dont le contrôleur aura besoin pour prendre ses décisions.

3.1 Environnement domotique de référence

Le système domotique à mettre en place dans le projet SWEET-HOME est composé de plusieurs éléments qui traitent et échangent des informations lors de l'interaction du système avec l'utilisateur. Ces éléments sont dédiés aux différentes tâches requises pour la création d'un environnement perceptif : l'obtention et le stockage des données, leur traitement, l'inférence des connaissances et le pilotage des actionneurs pour modifier l'état courant du système si nécessaire. Le composant central du système est le contrôleur intelligent qui implémente des méthodes d'intelligence artificielle pour raisonner et prendre les décisions plus appropriées pour l'habitant considérant sa situation. En conséquence, il est l'élément essentiel qui permet l'implémentation de l'intelligence ambiante dans le système domotique. Nos recherches sont donc orientées vers la réalisation de ce contrôleur.

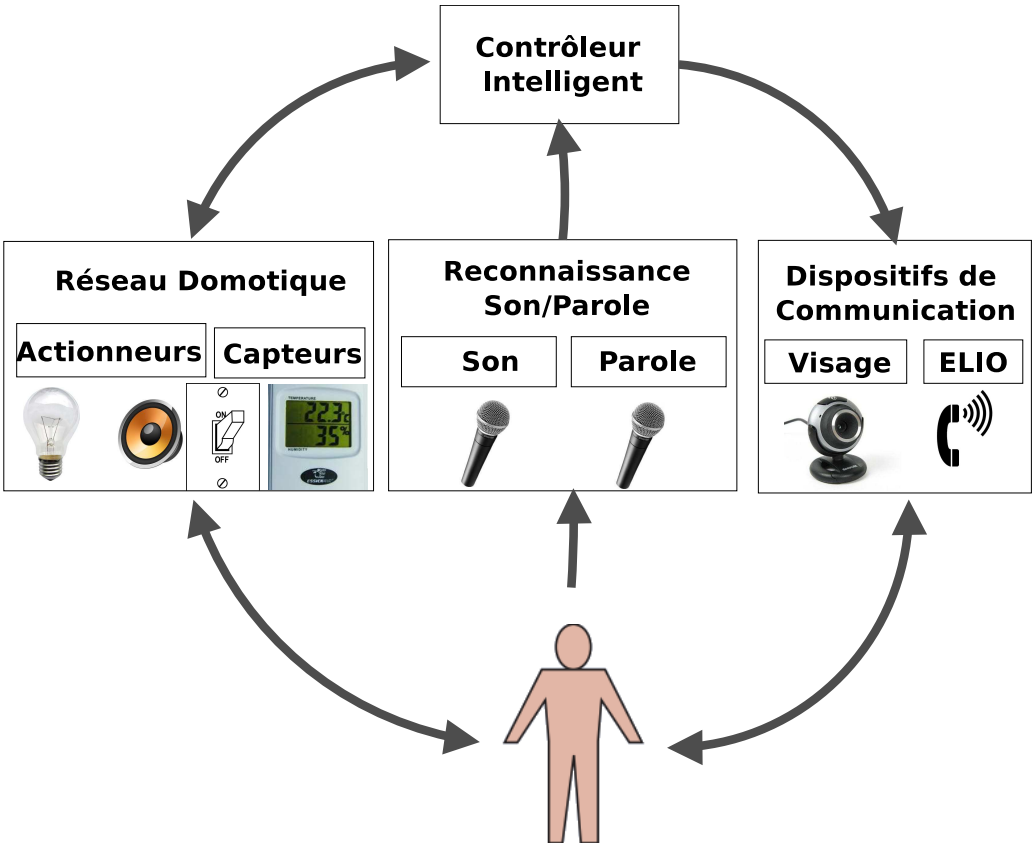


FIGURE 3.1: Interactions dans le système domotique

La figure 3.1 montre la façon dont est organisée l'interaction entre l'habitant, le contrôleur et les autres composants du système domotique. Étant donné que l'inférence d'in-

formations à partir des données brutes n'est pas possible, le contrôleur met à profit les informations provenant du réseau des capteurs domotiques et du système de reconnaissance son/parole. À partir de ces informations, il génère les messages à envoyer à l'utilisateur par l'intermédiaire du synthétiseur vocal ou les ordres à destination des actionneurs dans la maison intelligente. Le contrôleur peut aussi mettre en contact l'habitant avec ses proches ou émettre des appels d'urgence d'une façon proactive s'il évalue que l'habitant se trouve dans une situation de risque en s'appuyant sur des systèmes de communication avec l'extérieur spécialisés comme e-lio ou Visage. Le système de reconnaissance son/parole discrimine tout d'abord les signaux audio en deux classes, son ou parole ; ensuite, selon le résultat de la discrimination, il procède à la reconnaissance de la parole prononcée ou fait une classification du son sur un ensemble de classes de sons environnementaux. Les sons de la vie courante peuvent être une source d'information très riche pour lever l'ambiguïté dans l'analyse du contexte, par exemple la reconnaissance de sons de vaisselle dans la cuisine peut aider à identifier l'activité de rangement. Cependant, la reconnaissance de parole est la plus importante car l'interaction entre l'habitant et le contrôleur intelligent se fait principalement par des ordres vocaux.

Nous allons maintenant présenter en détail les informations que le contrôleur utilise pour exécuter les ordres commandés par l'habitant et pour inférer des situations menant à l'exécution d'une commande domotique. Nous décrirons ensuite les informations de sortie qui sont envoyés par le contrôleur à destination des autres sous-systèmes, telles que les alertes ou le changement d'état des actionneurs.

3.1.1 Informations d'entrée

Les informations d'entrée pour le contrôleur intelligent proviennent des messages transitant sur le réseau domotique KNX et des sorties du système d'analyse audio. Le tableau 3.1 liste les informations d'entrée.

Les informations reçues du réseau KNX sont de natures différentes. Certaines prennent des valeurs binaires $\{0;1\}$ ou $\{-1;1\}$, d'autres indiquent le niveau d'un réglage et prennent des valeurs entières sur une plage limitée $[0;100]$, d'autres mesurent un paramètre physique

Type	Capteurs	Type d'information	Nombre de capteurs
Présence	Infra-rouge (PIR)	Valeur binaire (0/1)	4
Porte/fenêtre	Contact binaire	Valeur binaire (0/1)	15
Volet/rideaux	Contact binaire	Valeur binaire (-1/1)	7
Intensité d'éclairage	Interrupteurs	Valeur numérique entière (0-100)	10
Température/CO_2	Détecteur	Valeur numérique à intervalle régulier	2
Débit eau/électricité	Compteur	Valeur numérique par pallier	1
Son	Microphone	Classe de son et Rapport Signal sur Bruit (RSB)	8
Parole	Microphone	Ordre/appel (chaîne ASCII) et RSB	8

TABLE 3.1: Informations en entrée

et sont exprimée par une valeur continue quantifiée.

3.1.2 Informations de sortie

Les sorties du contrôleur sont de trois types : les commandes domotiques générées sur le réseau, les messages à l'attention de l'utilisateur et les messages envoyés aux systèmes spécialisés.

3.1.2.1 Commandes domotiques

Ces commandes permettent de changer l'état d'un actionneur dans l'appartement intelligent. Une commande peut être déclenchée soit directement par l'habitant à travers les interfaces tactiles usuelles, soit par le contrôleur intelligent grâce à la reconnaissance d'une commande vocale ou par la détection d'une situation particulière. La liste des actionneurs qui peuvent être actionnés par une commande est affichée dans le tableau 3.2. Quelques actionneurs fournissent aussi des informations d'entrée quand ils sont activés manuellement par l'habitant. Le tableau 3.3 montre les différents types de messages qui peuvent être émis par le contrôleur.

3.1.2.2 Message vocaux

Les messages à l'utilisateur sont un moyen d'avertir l'utilisateur d'un danger (p.ex., plaques à gaz allumées depuis trop longtemps), d'un oubli (p.ex., téléphone non raccroché) ou d'un événement à venir (p.ex., un rendez-vous), etc. Le tableau 3.3 à la page 61 donne une liste d'exemple de messages. Chaque message est précédé d'une petite musique pour avertir que le système va émettre un message. Cette introduction sonore est un souhait du groupe d'utilisateurs identifié lors de l'étude d'usage (Portet et coll., 2013).

3.1.2.3 Messages à destination des systèmes spécialisés

Ces messages déclenchent la mise en communication avec les proches en utilisant des systèmes spécifiques de lien avec l'extérieur tels que e-lio ou Visage. La mise en contact peut être demandée par l'habitant mais le contrôleur peut aussi décider de l'exécuter directement si une situation de détresse est inférée.

Type	Effecteurs	Type d'information	nombre
Volet/rideaux	Contact binaire	Valeur binaire (-1/1)	7
Appareils électriques	Prise électriques	Valeur binaire (0/1)	10
Éclairage	Lumières	Valeur numérique discrète(0-100)	10
Message	Synthétiseur vocal	Texte ASCII	1
Communications	e-lio	Requête HTTP	1

TABLE 3.2: Sorties vers les actionneurs

3.2 Scénarios

Pour guider le développement du contrôleur intelligent, un certain nombre de scénarios ont été conçus afin d'illustrer son fonctionnement.

Scénario 1 *L'habitant se réveille au milieu de la nuit. Plutôt que de chercher l'interrupteur, il demande de la lumière. Le système allume les lampes de chevet de manière progressive. L'habitant se lève, sort dans le couloir et va aux toilettes, la lumière s'allume automatiquement sur son chemin. L'habitant retourne dans la chambre, les lumières s'éteignent derrière lui. Une fois dans son lit, il demande l'extinction des lumières et se rendort.*

Scénario 2 *Il est 21 heures, l'habitante se prépare à aller au lit. Elle va dans son lit et lit un livre. Au bout de 20 minutes, le système lui signale que la porte d'entrée n'est pas verrouillée. Elle pense qu'elle le fera plus tard puis 5 minutes plus tard elle éteint la lumière. le système lui signale que la porte d'entrée n'est pas verrouillée. Elle demande la lumière, va verrouiller la porte, retourne dans la chambre, éteint la lumière et s'endort.*

Scénario 3 *L'habitant entre dans l'appartement le soir, et, le contrôleur infère à ce moment là qu'il faut allumer la lumière de la première pièce. Ensuite, l'habitant prépare un plat pour son repas et le laisse sur la plaque électrique de la cuisine ; au même moment, il demande par un ordre vocal d'allumer la radio. Il se dirige ensuite à son bureau pour utiliser l'ordinateur. Après quelques minutes, il reçoit un*

Type	Description
Rappel	Des messages du type alerte qui ont été programmés, par exemple la prise d'un médicament : « *Vous avez oublié de prendre votre médicament* »
Situation dangereuse	Pour prévenir l'habitant lorsqu'une situation dangereuse est inférée. Éventuellement le message peut être suivi d'une mis en contact avec un proche de l'habitant : « *Vous avez oublié de fermer la porte d'entrée* »
Demande de confirmation	Après la réception d'une commande vocale, le contrôleur peut demander une confirmation : « *Voulez vous baisser le chauffage ?* »
Conseil	Messages dans des situations où l'action à effectuer par l'habitant n'est pas très importante. Par exemple, si la température est basse : « *Vous feriez bien de fermer la fenêtre* »
Confirmation d'une action invisible	————-

TABLE 3.3: Messages vocaux à destination de l'habitant

appel téléphonique et le contrôleur diminue immédiatement le volume de la radio. Une demi heure après, le contrôleur alerte l'habitant au moyen d'un message vocal pour l'informer qu'il a laissé la plaque allumée.

Scénario 4 *Il est 14 heures, l'habitante se prépare à aller en ville. Elle ouvre la porte mais est interrompue par le système qui lui signale que la fenêtre est ouverte. Elle laisse la fenêtre ouverte car elle n'en a que pour 5 minutes et souhaite aérer.*

Scénario 5 *Il est 15 heures, l'habitante se réveille de la sieste et va au toilette. Elle glisse et tombe violemment sur le sol. Elle n'arrive pas à se relever et le téléphone est dans une autre pièce. Elle demande de l'aide au système qui appelle sa fille. N'ayant pas de réponse, le système appelle les secours. Les secours décrochent et peuvent communiquer avec l'habitante à travers les micros et les haut-parleurs du système domotique.*

3.3 Inférence et hiérarchie des informations

Les informations d'entrée sont traitées par les modules d'inférence du contrôleur afin d'obtenir des informations ayant un niveau sémantique plus significatif ; ensuite, les résultats de ces inférences servent d'entrée aux autres modules donnant ainsi lieu à une organisation hiérarchique de la production de l'information. La figure 3.2 montre la façon dans laquelle ces informations sont organisées par rapport à leurs sources. Les commandes tactiles et les ordres vocaux, préalablement traités par le système de reconnaissance vocale, sont traités directement par le module de décision une fois qu'elles sont reconnues. En fait, la détection d'un ordre vocal est en soit une condition suffisante pour analyser l'état courant du système et décider de l'action la plus convenable pour la personne. Par contre, la reconnaissance de situations précises est d'un traitement plus complexe, car l'occurrence d'un évènement déclenché par un capteur domotique n'est pas suffisante en elle-même pour déterminer que l'on se trouve dans une circonstance qui nécessite l'analyse de l'état du système en vue d'une prise de décision. Le dernier niveau de cette hiérarchie correspond à l'émission des informations de sortie résultant de la prise de décision afin de répondre à un ordre de l'utilisateur ou une situation particulière.

3.4 Architecture fonctionnelle du contrôleur intelligent

Toutes les actions qui devront être prises pour garantir le confort et la sécurité de l'habitant vont résulter des processus d'inférence exécutés par le contrôleur intelligent. Le contrôleur utilise les autres composants du système pour obtenir des informations convenablement traitées et pour envoyer des commandes dont l'effet sera de modifier l'environnement au profit de l'habitant. Ainsi, le contrôleur devient l'élément central du système domotique.

Le logiciel implémentant le contrôleur, développé en langage Java, est composé de modules dédiées à l'inférence, consacrés chacun à un aspect en rapport avec la prise de décision

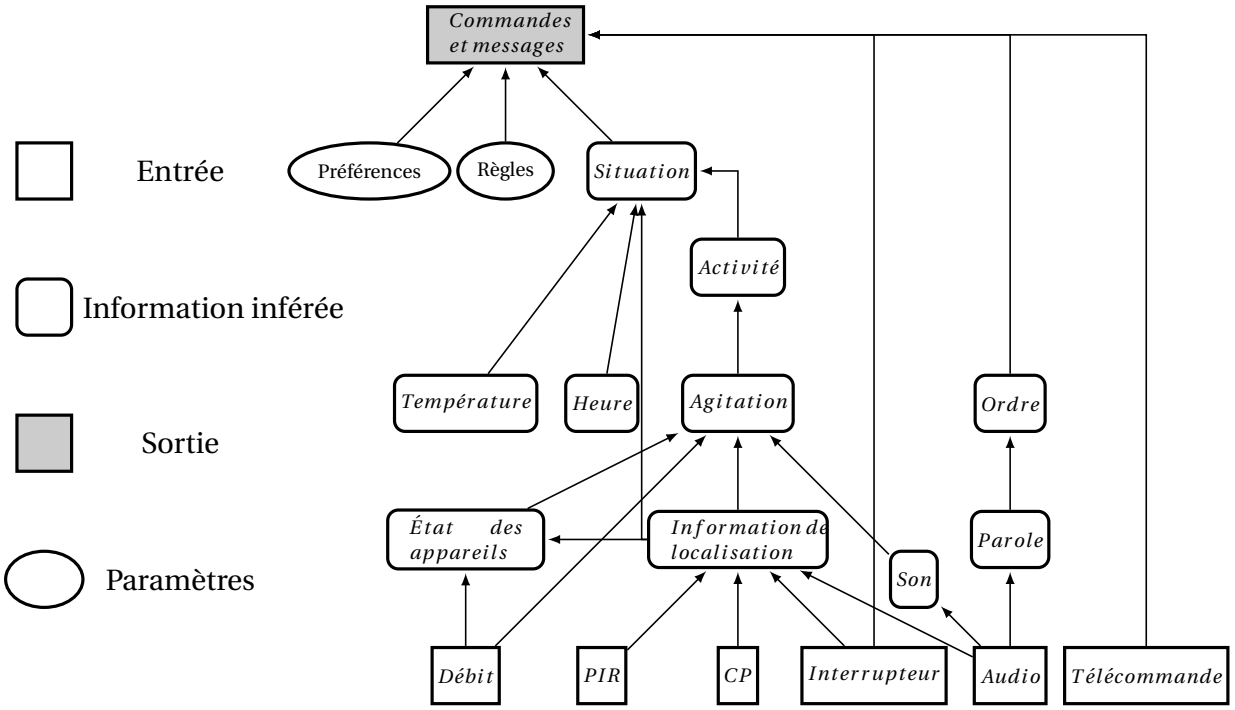


FIGURE 3.2: Informations intermédiaires inférées à partir informations d'entrée (PIR=détecteur infra-rouge, CP=contact de porte)

(localisation et reconnaissance d'activité), la communication avec des systèmes externes, et l'interaction avec la base de connaissance. Cette section détaille l'architecture du contrôleur intelligent, l'interaction entre ses différents composants, et l'interaction avec des systèmes externes.

3.4.1 Structure générale

Les composants du contrôleur intelligent sont détaillés sur la figure 3.3. Ils sont organisés principalement en trois groupes : les interfaces, les modules d'inférence et la base de connaissance. Les modules du premier groupe permettent la communication avec les systèmes externes du contrôleur et le pré-traitement des données pour leur exploitation. Chaque module d'inférence met en œuvre un algorithme pour inférer une information de haut niveau utile pour la reconnaissance de situation. La base de connaissance contient une représentation des éléments appartenant à l'environnement perceptif mais aussi les observations de bas et haut niveau (celles obtenues par les modules d'inférence) qui sont stockés ici dès qu'ils sont détectées. Le schéma montre l'organisation hiérarchique des modules. Par exemple pour inférer une activité, il est nécessaire de connaitre la localisation de l'habitant. Après l'arrivée d'une observation de bas niveau, les informations immédiatement supérieures dans la hiérarchie sont inférées par les modules correspondant et ainsi de suite pour les autres niveaux. Les derniers composants du processus sont la reconnaissance de situation et la prise de décision dans le cas où une action doit être exécutée. Toutes les observations sont prises en compte pour déterminer si le changement dans l'état du système est synonyme de situation à risque.

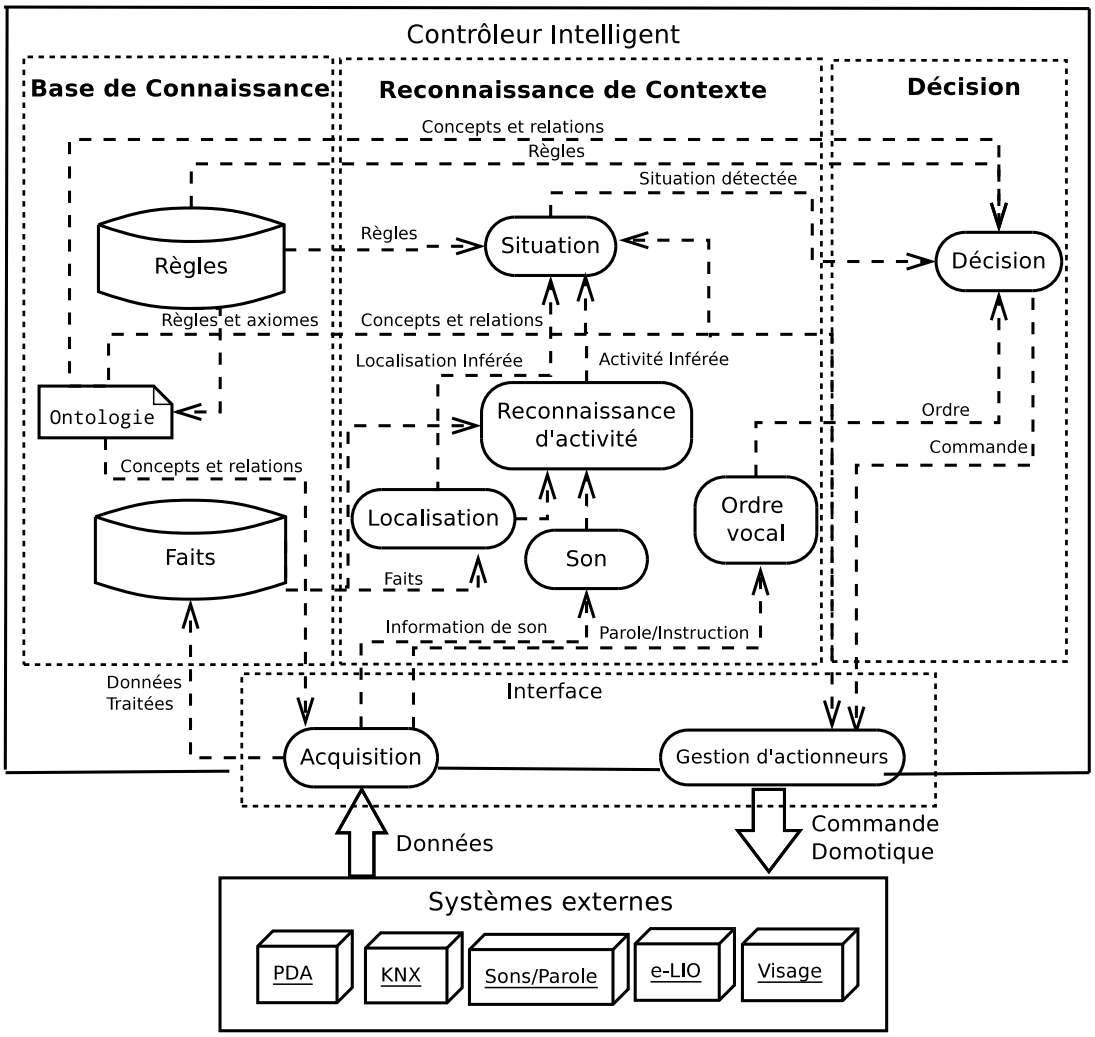


FIGURE 3.3: Modules du contrôleur intelligent

Les modèles d'inférence du contrôleur traitent des éléments essentiels pour la désambiguïsation du contexte, tels que la localisation et la reconnaissance d'activité. Nous allons maintenant faire une présentation succincte de chacun des modules.

3.4.1.0.1 Ontologie

Certaines connaissances concernant l'appartement et l'habitant peuvent être spécifiées avec une ontologie. Il s'agit d'une représentation formelle des connaissances au travers de concepts appartenant à un domaine mais aussi de leurs relations ; des instances de ces concepts peuvent aussi être spécifiées pour faire des inférences sur les propriétés du domaine. Les avantages principaux d'une ontologie sont un vocabulaire standard pour décrire le domaine et la facilité pour faire des modifications dans le contenu quand cela est nécessaire. Parmi les connaissances qu'il est possible de spécifier, nous pouvons citer par exemple la localisation des pièces dans l'appartement, la localisation des capteurs, les caractéristiques de la personne (âge, déficiences, préférences). L'ontologie peut être utilisée aussi pour faire une abstractions des informations de KNX vers éléments logiques qui sont utilisés pour le reconnaissance de contexte.

La représentation ontologique que nous avons élaborée est détaillée dans la section 3.5

3.4.1.0.2 Acquisition de Données

Le rôle de ce module est d'enregistrer les données qui transitent sur le bus KNX ou qui sont émises par le système de reconnaissance de son/parole et de les mettre dans un format qui puisse être utilisable directement par les modules d'inférence. Ici, l'ontologie est utilisée par le module pour obtenir des informations telles que la correspondance entre un capteur particulier et une pièce de l'appartement.

La solution est de stocker les données comme instances d'atomes logiques, c'est à dire des prédicats logiques où les paramètres sont des constantes. Par exemple, il est possible d'enregistrer une détection de présence dans la cuisine par un capteur infra-rouge PIR à un instante donnée sous la forme $PIR(cuisine, 16:05:01)$. Ce module est donc en charge du pré-traitement des données.

3.4.1.0.3 Base de Faits

Les informations obtenues des capteurs sont stockées dans ce module. Comme mentionné au dessus, il peut s'agir d'une base de prédicats logiques. Pourtant cela peut aussi être un simple fichier de texte contenant les données qui servent d'entrée aux méthodes de raisonnement. Dans notre implémentation, cette base est intégrée à l'ontologie. Il faut noter que pour faire un raisonnement temporel, il est nécessaire que toutes les données reçues dans un intervalle de temps soient stockées, ainsi cette base est une sorte de mémoire. Par conséquence, ces informations sont étiquetées avec le temps d'occurrence.

3.4.1.0.4 Détection de Localisation

Il s'agit du module qui implémente la méthode pour la localisation de l'habitant dans l'appartement à un instant donné en utilisant les réseaux dynamiques. Le fonctionnement de ce module est expliqué dans la section 4.3.

3.4.1.0.5 Reconnaissance d'activité

Le but de ce module est de déterminer l'activité de vie quotidienne réalisée par l'habitant dans un instant donnée. Le fonctionnement de ce module est détaillé dans la section 4.4.

3.4.1.0.6 Reconnaissance d'ordres

Les sons qui sont classifiés comme parole sont analysés directement par le système de Reconnaissance Automatique de la Parole (RAP), le résultat obtenu est traité dans ce module où une évaluation est faite pour déterminer s'il s'agit bien d'un ordre à exécuter pour le contrôleur. Les critères pour classifier les ordres et les commandes à exécuter sont spécifiés dans la base de règles.

3.4.1.0.7 Base de Règles

Dans le processus d'inférence de contexte et de prise de décision, il est important de pouvoir inclure des connaissances explicites au delà des connaissances apprises par apprentissage automatique. Par exemple, des règles logiques pour décrire que si la personne se trouve quelques minutes dans la cuisine et que, dans cet intervalle, elle a allumé le four et ouvert le frigo, alors l'activité en cours est « faire la cuisine ». Si une méthode de programmation logique statistique est appliquée, une telle base peut avoir un poids numérique associé à chaque règle pour modéliser l'incertitude. Alors cette valeur représente la probabilité qu'une règle soient vraie dans le monde réel.

Les règles de ce module servent surtout à spécifier les situations d'intérêt et les actions qui doivent s'exécuter quand elles sont reconnues. La section 4.5 décrit en détail ce module.

3.4.1.0.8 Détection de Situations

La détermination des situations est faite à partir des informations sur l'état du système mais en utilisant des informations complémentaires qui ont été inférées, à savoir, la localisation, le temps, et l'activité en cours de réalisation (nous considérons que la reconnaissance de l'activité est faite lors de la reconnaissance de situations). La méthode choisie est basée sur le raisonnement sur des ontologies (se reporter à la section 4.2.2.6 pour plus de précisions).

3.4.1.0.9 Prise de décision

Ce composant prend comme entrée la sortie du module de reconnaissance du contexte pour décider s'il faut agir ou non, et si il il y lieu de l'action qui doit être entreprise. Éventuellement cela se traduira pour le module suivant à envoyer des commandes à des actionneurs en lien avec les appareils de l'appartement : allumer la lumière, éteindre le four, faire un appel d'urgence.

3.4.1.0.10 Gestion des actionneurs

Après la prise de décision, les actions exécutées par le contrôleur doivent se traduire en ordres vers les dispositifs à travers un module qui interagit avec eux. Si, par exemple, il faut allumer la lumière dans une pièce, il faut indiquer sur le bus KNX quels sont les dispositifs précis à allumer. Dans cette étape, on peut se servir des connaissances présentes dans l'ontologie. L'ontologie pourrait aussi décrire les actions à réaliser, telle que les étapes qui constituent chaque action.

3.4.2 Architecture du logiciel

Le contrôleur a été implémenté entièrement en langage Java. L'utilisation d'un langage de programmation *orienté objet* est très pertinent dans nos recherches à cause de la modularité que l'implémentation du contrôleur nécessite et aussi de la facilité à ajouter des composants optionnels. Toutes les classes utilisent le patron de conception *Singleton* (Gamma

et coll., 1995) pour restreindre leur instanciation à un seul objet qui reste en mémoire pendant tout l'exécution du système. En fait, toute la chaîne de traitement est coordonnée avec seulement une instance de chaque classe lors de l'arrivée d'un événement, donc il n'existe pas un instant t dans lequel plus d'un instance est nécessaire. Ce modèle améliore la performance du système puisqu'il requiert moins de mémoire que la génération des instances dans chaque évènement. La bibliothèque OWLAPI (Horridge et Bechhofer, 2011) a été employée pour manipuler les ontologies dont le langage est OWL2. Le deuxième composant externe intégré au contrôleur est la bibliothèque *Carados* utilisée pour consulter les informations stockées dans l'ontologie de façon efficace.

La figure 3.4 montre le diagramme de classes et l'organisation en paquetage du contrôleur. Le premier paquetage contient les classes qui permettent la connexion avec les sources d'information et les actionneurs. Le paquetage principal regroupe les classes qui implémentent l'accès à la base de connaissance et les méthodes d'inférence. Finalement, le logiciel s'appuie sur un paquetage pour les patrons de conception dont le principal est *l'observateur* (Gamma et coll., 1995). L'inférence des informations hiérarchiques est implémentée à travers ce patron. Ainsi, les classes en charge d'inférer des informations de haut niveau sont des observateurs et les classes successeurs dans la hiérarchie sont des objets d'observation.

La liste suivante décrit les classes qui font partie du contrôleur :

- **DataProvider :** cette interface déclare les méthodes pour accéder à une source d'information.
- **WebServerClient :** classe qui implémente l'interface *DataProvider* pour effectuer la connexion avec le service web de KNX.
- **FileReader :** classe qui implémente l'interface *DataProvider* pour effectuer la lecture d'un fichier texte. Le système de reconnaissance son/parole, par exemple, pourra écrire ses informations dans un fichier permettant sa prise en compte par le contrôleur.
- **Observable :** Une classe qui implémente cette interface possède une liste d'objets *Observer* de façon telle que lors de l'occurrence d'un événement spécial (l'arrivée d'informations sur KNX, par exemple), l'objet *Observable* parcourt sa liste pour envoyer une notification à tous ses objets *Observer*.
- **Observer :** Interface implémentée par des classes qui attendent des notifications des objets *Observable*.
- **InputManager :** Classe qui s'occupe du pré-traitement des données reçues par les systèmes externes.
- **OntologyManager :** Cette classe contient les méthodes pour interagir avec la base de connaissance. Elle utilise des API's pour traiter le langage OWL et des composants externes pour effectuer le raisonnement sur l'ontologie.
- **DataProducer :** Classe abstraite avec des méthodes pour faire des inférences d'information de haut niveau. Ses classes concrètes permettent d'inférer la localisation ou l'activité courante, par exemple. Un cas spécial est la classe *Decision* qui est considérée aussi comme un *DataProducer* parce qu'elle est la dernière dans la hiérarchie pouvant inférer des informations de haut niveau tout en recevant la notification des détections

de situations.

– **Command :** Les méthodes implémentées dans cette classe permettent d'envoyer des commandes domotiques vers le bus KNX ou bien vers les dispositifs spécialisés pour la mise en communication de l'habitant avec ses proches.

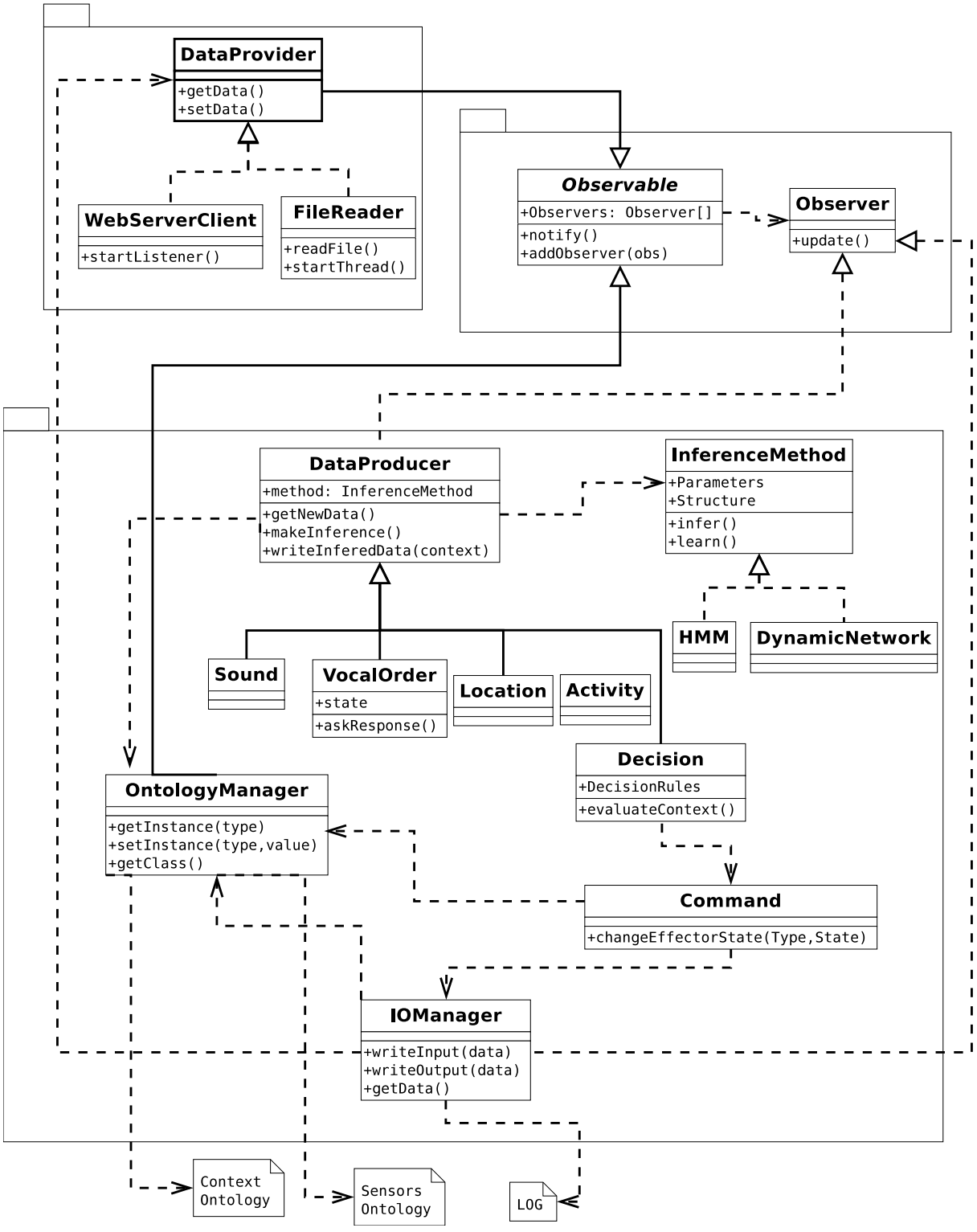


FIGURE 3.4: Diagramme de classes du contrôleur intelligent

3.4.3 Interaction entre modules

L'interaction entre les modules du contrôleur est rendue possible par l'implémentation des interfaces *Observer* et *Observable* qui établit un ordre dans tout le processus depuis l'acquisition des données jusqu'à l'exécution des commandes domotiques. L'élément principal dans la communication entre modules est la base de faits puisque elle stocke toute information inférée par un module et le rend disponible pour les modules consécutifs. Ainsi, l'accès à la base de faits suit une approche orientée services qui permet d'inclure un modules d'inférence et le configurer pour être notifié seulement à un ensemble particulier des événements. En plus, tous les modules partagent la même structure de données et la même définition des connaissances relatives à l'environnement.

3.4.4 Interaction avec les systèmes externes

La communication avec les systèmes externes est faite par l'utilisation de protocoles de réseau standard, tel que HTTP afin de garantir la facilité de maintenance du système car cela permet de suivre plus facilement les problèmes de communication, de modifier les interfaces après une mise à jour du matériel ou le changement d'un système externe. Outre la facilité de maintenance, l'interopérabilité est une propriété indispensable au cas où il faut adapter le contrôleur intelligent à un nouvel environnement perceptif.

3.5 Représentation des connaissances

Les connaissances nécessaires pour les opérations du contrôleur intelligent sont organisées selon deux couches sémantiques : une ontologie de bas niveau et une ontologie de haut niveau. La séparation des représentations ontologiques dans différents niveaux d'abstractions a été souvent appliquée en intelligence ambiante (Gu et coll., 2005; Wolf et coll., 2008) afin de rendre les systèmes domotiques plus modulables. La ségrégation entre les connaissances concernant les processus d'inférence et celles représentant la structure physique de l'environnement permet utilement de diminuer l'impact de changements de configuration dans le réseau domotique sur les processus d'inférence. En plus, cela permet d'adapter facilement le système à d'autres maisons intelligentes ayant une infrastructure différente, et même à d'autres environnements intelligents, salles de classe par exemple.

3.5.1 Ontologie de bas niveau

Dans notre implémentation, l'ontologie de bas niveau est dédiée à la représentation des éléments qui composent l'environnement perceptif. À ce niveau, nous nous intéressons tout d'abord à définir les régions de l'espace physique, les lieux et leurs subdivisions, ensuite le placement des objets dans l'environnement physique. Ce peut être des objets qui ne sont pas attachés à une localisation spécifique comme les fenêtres, ou d'autres qui sont caractéristiques d'une certain localisation comme le réfrigérateur qui est généralement dans la

cuisine. Cette distinction est essentielle pour déduire des informations de haut niveau (localisation et activité) à partir des évidences sur l'utilisation de certains de ces objets. L'ontologie décrit aussi les dispositifs associés aux objets, c'est à partir des capteurs que le contrôleur peut repérer dans certains cas la manipulation d'un objet. Les informations inscrites dans cette ontologie incluent le type d'information fournie par les dispositifs, leurs états possibles, et leurs identifiants. Ces connaissances là ne changent pas lors du fonctionnement du système. Cependant, les changements de l'état de ces dispositifs sont eux aussi stockés entièrement dans cette ontologie. Le but poursuivi est donc de construire une représentation de l'espace physique qui servira au stockage des données temporelles de bas niveau.

3.5.2 Ontologie de haut niveau

L'ontologie de haut niveau représente des concepts qui sont utilisés au niveau raisonnement. Ces concepts s'organisent en trois parties listées ci-dessous :

- *Entité abstraite*. Ces concepts concernent les éléments utilisés dans les inférences qui n'ont pas relation avec des objets physiques, tels que les concepts *Situation* ou *Commande*. Dans la plupart des cas, les instances de ces concepts sont les valeurs possibles résultant d'un processus d'inférence. Ainsi, par exemple, *cuisiner* ou *dormir* sont des instances appartenant à la classe *Activité*.
- *Entité physique*. Cela regroupe les classes qui désignent des objets physiques que l'on trouve dans l'espace physique ou les pièces existantes. À la différence de l'ontologie de bas niveau, ici on ne cherche pas à décrire les objets spécifiques dans l'environnement, mais plutôt à lister les objets existants de l'appartement sans s'intéresser à des propriétés spécifiques comme leur localisation ou identifiants. Par exemple, une instance de la classe *Objet* est simplement *porte*, sans spécifier à quel objet réel dans l'espace elle se réfère.
- *Évènement*. Les instances dans l'ontologie de haut niveau sont produites par les modules d'inférence (ex. localisation, activité, et situation) après traitement des informations provenant des capteurs (qui ont été préalablement stockées dans l'ontologie de bas niveau). Ces instances correspondent à la concrétisation des entités abstraites, comme la réalisation d'une activité dans une localisation et à un instant précis (par exemple, à 13h :00 l'habitant prend son repas à la cuisine).

3.5.3 Relation entre les deux ontologies

La figure 3.5 montre quelques concepts et relations des deux ontologies. L'Abox sert comme exemple de l'état de la base de faits à un instant donné. Les faits décrivent l'état de quelques éléments lors du déroulement du **Scénario 1** décrit dans la section 3.2 quand l'habitant éteint la lampe de chevet avant de dormir. Le contrôleur met à jour l'état des dispositifs dans l'ontologie de bas niveau, et il peut être inféré — encore au bas niveau — que toutes les lumières dans la chambre sont éteintes. Dans l'ontologie de haut niveau, l'interaction avec la lampe est stockée comme un événement de dispositif ayant l'instant d'occurrence

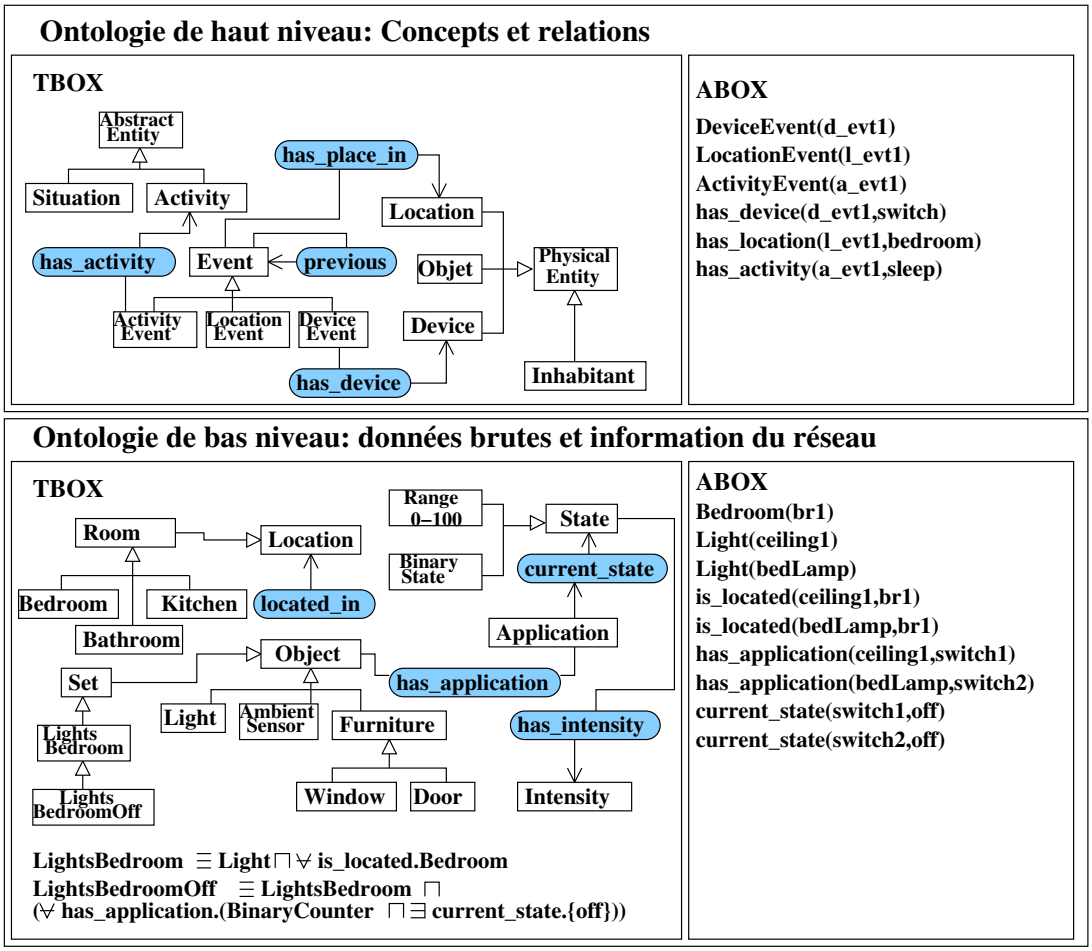


FIGURE 3.5: Ontologies représentant l'environnement perceptif

et la pièce comme propriétés. Ensuite, le module de localisation est employé et le résultat de la localisation est directement la chambre puisque l'interrupteur se trouve dans cette pièce. Les évidences sur la localisation de l'habitant, l'état des lumières, et la période de la journée, peuvent suffire pour inférer que l'activité courant est dormir. Finalement, ces inférences fournissent le contexte sur lequel la reconnaissance de situation est faite. Dans le même scénario, si la personne oublie de fermer la porte principale et si une situation a été définie pour ce cas, cette situation sera étiquetée dans l'ontologie comme reconnue puis traitée par le module de décision. La reconnaissance de situation par le raisonnement avec les ontologies est décrite en détail dans la section 4.5.

Outre la représentation des connaissances, les ontologies offrent aussi la possibilité d'inférer des connaissances par le raisonnement logique, comme nous le détaillerons dans la section 4.2.2.5. Nous utilisons cette capacité pour analyser l'état de l'environnement à un certain instant par rapport à un ensemble d'instances. Dans l'ontologie de bas niveau, le concept *LightsBedroom* regroupe toutes les lumières qui se trouvent dans la chambre, alors que le sous-concept *LightsBedroomOff* désigne le sous-ensemble des lumières éteintes. Ces groupes sont très utiles pour la définitions de situations. Par exemple, pour définir une situation dans laquelle toutes les lumières de la chambre sont éteintes, par exemple quand la personne dort la nuit, il n'est pas nécessaire d'évaluer l'état de toutes les lumières, mais il

suffit d'évaluer s'il existe ou non des instances dans la classe *LightsBedroomOn*.

3.6 Synthèse

Ce chapitre a présenté la réalisation du contrôleur intelligent conçu pour piloter l'environnement perceptif. Nous avons d'abord présenté le système de domotique global, ses composants, les interactions, et le rôle tenu par le contrôleur. Cette description est importante pour comprendre quelles sont les données d'entrée et de sortie, ainsi que les interfaces nécessaires pour faciliter la communication. En suite, nous avons abordé l'architecture fonctionnelle du contrôleur qui se base sur une organisation hiérarchique des modules d'inférence qui transforme les données brutes en informations de haut niveau pour reconnaître des situations particulières et prendre des décisions. Finalement, nous avons détaillé et justifié les choix faits en ce qui concerne l'architecture du logiciel.

La conception du contrôleur intelligent prend en compte des propriétés essentielles de génie logiciel, c'est-à-dire la facilité de maintenance, l'extensibilité, et l'interopérabilité, de façon que le système résultant puisse être non seulement facilement modifiable selon l'évolution des dispositifs présents dans la maison intelligente mais puisse également être adapté pour son utilisation dans d'autres types d'environnement intelligents comportant des tâches de raisonnement et prise de décision similaires.

CHAPITRE 4

Inférence des informations

Dans les maisons intelligentes, l'abstraction de données consiste à extraire à partir des données de capteurs, des événements et des informations qui sont pertinents pour les systèmes domotiques de décision. Ceci implique d'abstraire les signaux bruts en symboles (propositions) étiquetés temporellement (p. ex. : claquement de porte à 11h32), à partir desquelles des signatures de situations particulières d'événements atomiques sont détectées à travers un processus d'abstraction disposé en couches. Par exemple, les mouvements détectés dans la cuisine peuvent faire partie de l'activité « prendre une casserole » qui elle même est en rapport avec la préparation du repas. L'inférence d'information de haut niveau appliquée aux environnements perceptifs est un des secteurs de recherche les plus actifs et les plus ambitieux en raison de la quantité importante de bruit dans les données et de leur degré d'incomplétude important.

Les informations d'entrée à partir desquelles le contrôle intelligent de la domotique est effectué proviennent d'entités que nous appelons *sources d'information* et qui sont soit :

1. les observations des capteurs domotiques installés dans l'environnement perceptif ou
2. les hypothèses du système de reconnaissance automatique de la parole et des sons. Ce sont ces données qui donnent au contrôleur intelligent sa capacité de perception de l'environnement.

Néanmoins, ces données d'entrée ne peuvent être utilisées directement pour caractériser la situation de l'habitant à un instant donné. Pour prendre une décision, le contrôleur doit pouvoir évaluer le contexte et la situation de la personne. Par exemple, les données d'entrée ne fournissent pas directement l'activité en cours ni n'indiquent explicitement si la personne se place dans une situation dangereuse. Il est donc important de mettre en place des modèles d'inférence permettant d'extraire ces informations de haut niveau des flux de données d'entrée. Comme pour toutes données provenant de capteurs utilisés dans une application réelle, un certain nombre de difficultés sont à prendre en compte avant d'effectuer cette tâche. Les données des capteurs domotiques et provenant du traitement audio peuvent être : incertaines, incomplètes, conflictuelles ou imprécises. Par ailleurs, les modèles d'inférence doivent également considérer la dimension temporelle. Par exemple, la détermination de la localisation courante peut être améliorée si le modèle d'inférence utilise les informations des localisations précédentes.

Une autre difficulté dans cette modélisation du système d'inférence est que les informa-

tions à extraite sont toutes d'un niveau sémantique différent. Comme illustré sur la figure 3.2, les informations utilisées par le contrôleur sont organisées d'une façon hiérarchique. Il existe donc d'une part, une nécessite d'adapter les méthodes d'inférence aux particularités des informations sources et d'autre part, une relation de dépendance entre les divers processus d'inférence. Par exemple, l'activité réalisée par l'habitant ne peut pas être inférée sans avoir inféré préalablement sa localisation.

Ce chapitre décrit les méthodes mises en œuvre pour faire l'inférence des éléments nécessaires à la mise à jour du contexte de décision. Parmi les éléments inférés les trois plus importants sont présentés. Il s'agit de la **localisation**, l'**activité** et la **situation** courante de la personne. Chaque processus d'inférence des éléments précités doit prendre en compte certaines caractéristiques des données d'entrées. Ces caractéristiques sont introduites en section 4.1. Les méthodes d'inférences les plus pertinentes de la littérature sont ensuite introduites en section 4.2. Cet état de l'art est organisé selon les familles de modèles d'inférence : statistiques, logiques, ou statistique-relationnels. Les méthodes de localisation de l'habitant dans l'environnement perceptif et de reconnaissance des activités de la vie quotidienne sont abordées dans les sections 4.3 et 4.4. Enfin, la méthode développée pour la reconnaissance de situations est décrite dans la section 4.5.

4.1 Des données brutes aux informations de haut niveau

L'inférence au sens logique du terme est l'opération qui consiste à admettre une proposition en raison de son lien avec une proposition préalablement tenue pour vraie. Cependant, l'inférence au sens large consiste à admettre un fait en fonction de prémisses qui peuvent être des informations sous forme logiques, numérique ou des modèles probabilistes (inférence probabiliste). Dans le cadre de notre étude, l'inférence des informations de haut niveau (p. .ex., activité) doit être réalisée à partir d'informations de bas niveau (p. .ex., issues d'un capteur infra-rouge) et d'une connaissance *a priori* du modèle d'inférence (p. ex., modèle probabiliste). Le but est bien de prendre en compte les faits ayant lieu dans l'environnement domotique et leurs relations pour obtenir de nouvelles connaissances sur l'état courant de l'environnement. Les mécanismes utilisés pour inférer ces connaissances doivent considérer les caractéristiques des données brutes qui rendent difficile l'interprétation des informations sources. Parmi les considérations à prendre en compte dans un algorithme d'inférence, il convient de citer principalement :

- **La temporalité.** L'état présent des sources d'information de l'environnement ne suffit pas à inférer des connaissances complexes. Il est nécessaire de faire une analyse temporelle qui prend en considération l'évolution des informations. La temporalité rend indispensable l'utilisation d'un mécanisme de stockage des informations précédentes.
- **L'incertitude.** L'information fournie par les capteurs est souvent incertaine ou erronée. En général, cette caractéristique est présente dans la plupart des applications utilisant des données produites par le monde physique. Le modèle d'inférence doit donc être capable de prendre en compte cette incertitude lors de l'inférence d'un fait et de pro-

duire une estimation de l'incertitude de celui-ci.

- **L'hétérogénéité.** L'inférence doit être effectuée en utilisant des données très diverses. Celles-ci peuvent être scalaires ou non, numériques ou symboliques, discrètes ou continues, avoir un haut niveau sémantique ou non. Cette diversité complique grandement la tâche d'agrégation ou d'interprétation des données et il est souvent nécessaire de mettre en place des couches de pre-traitement afin de transformer ces données hétérogènes dans un mode de représentation permettant leur exploitation conjointe.
- **Le traitement en ligne.** L'inférence *en ligne* (c.-à-d., au fur et à mesure que les données arrivent en entrée du système) s'oppose au traitement *hors ligne* des données. Dans le traitement hors ligne, l'ensemble des données est disponible pour effectuer l'inférence. Dans le cas du traitement en ligne, seules les informations courantes et passées sont disponibles. Un traitement en ligne dispose donc de beaucoup moins d'information qu'un traitement hors-ligne et est souvent soumis à des contraintes de temps. Par exemple, dans le cas particulier des systèmes temps-réel, le temps de réponse d'un système d'inférence est strictement borné. Dans notre cadre, la décision du contrôleur doit être prise en un temps acceptable par l'utilisateur.
- **L'information conflictuelle.** Il est possible trouver des informations contradictoires en entrée d'un processus d'inférence. Par exemple, pour inférer une localisation, deux capteurs peuvent fournir des informations signalant une présence à des endroits différents. Dans ce cas, les méthodes d'inférence doivent embarquer un mécanisme de résolution de conflits qui peut consister en l'élaboration d'un degré de confiance pour chaque source d'information selon la situation courante.

En plus de ces considérations sur la nature des données et du traitement, toutes les méthodes d'inférence doivent également partager une base de stockage de données commune afin d'éviter les redondances, utiliser un seul format de données et une seule description sémantique. Dans notre travail, la description sera réalisée à travers une ontologie exprimée en logique de description. Les concepts décrivent des éléments relatifs à l'environnent et les instances correspondent aux données obtenues à partir des sources d'information. Par rapport aux schémas de description classiques (p. ex., DTD ou modèle entité-relation), cette approche permet de représenter des relations plus complexes entre les entités, de déduire des informations supplémentaires à partir des faits stockés, et finalement d'avoir un vocabulaire commun aux méthodes d'inférence pour éviter l'ambiguïté.

4.2 Méthodes d'inférence

Les méthodes décrites dans cette section ont été organisées selon les approches les plus courantes de l'intelligence artificielle : les méthodes statistiques parmi lesquelles nous considérons les modèles probabilistes, et les méthodes logiques qui se basent, pour la plupart, sur la logique du premier ordre. Enfin, nous considérons une troisième classe qui inclut les méthodes développées récemment dans la communauté de l'apprentissage automatique et qui combinent des modèles probabilistes et des règles logiques afin de tirer partie de la puis-

sance d'expressivité des représentations logiques et de la capacité des modèles probabilistes à traiter l'incertitude.

4.2.1 Approches statistiques

Les méthodes réunies sous cette dénomination sont celles qui permettent de traiter les imperfections associées aux données d'entrée. Il s'agit principalement de modèles de classification automatique pour lesquels il existe des méthodes d'apprentissage automatique pour apprendre ces modèles à partir de données. Le résultat de la classification devient une information de haut niveau. Dans cette section, l'accent est surtout mis sur les modèles bayésiens qui permettent de prendre en compte l'incertitude sous la forme d'une probabilité.

4.2.1.1 Les Séparateurs à Vaste Marge (SVM)

Les séparateurs à vaste marge (SVM) (Boser et coll., 1992; Cortes et Vapnik, 1995) sont des outils de classification et régression qui utilisent la théorie de l'apprentissage automatique pour maximiser l'exactitude de prédiction et éviter le sur-apprentissage. Un SVM peut être défini comme un système qui utilise des fonctions linaires dans un grand espace multidimensionnel, modélisé avec un algorithme d'apprentissage ayant pour base la théorie de l'optimisation (Boser et coll., 1992). Tout comme les réseaux de neurones, le SVM possèdent la capacité de pouvoir approximer n'importe qu'elle fonction multivariée avec un degré d'exactitude acceptable. Ils sont utiles pour modéliser des systèmes complexes linaires ou non linaires.

Dans le cas binaire, le problème de l'apprentissage d'un SVM consiste en la construction d'un hyperplan de séparation $\vec{w}^T\vec{x} + w_0 = 0$ qui sera ensuite utilisé par une fonction $f()$ qui permet de savoir à laquelle des 2 classes appartient un vecteur $\vec{y}$. Un exemple de fonction est donné en 4.1.

$$f(\vec{w}^T\vec{y} + w_0) = sign(\vec{w}^T\vec{y} + w_0) \tag{4.1}$$

Étant donné un ensemble de N exemples $\{(\vec{x}_1, e_1), \ldots, (\vec{x}_N, e_N)\}$ constituant le jeux d'apprentissage, où $\vec{x}_i$ sont les vecteurs de caractéristiques de dimension D, et $e_i \in \{-1, +1\}$ les classes, l'hyperplan choisi doit maximiser la distance entre les points les plus proches de chaque classe tout en restant un séparateur.

Cela revient à minimiser $\frac{1}{2}\|\vec{w}\|^2$ sous les contraintes $e_i(\vec{w}^T\vec{x}_i + w_0) \geq 1$, pour $i \in 1 \ldots N$. Ceci est classiquement résolu par la méthode des multiplicateurs de Lagrange, où le lagrangien est donné par l'equation :

$$L(\vec{w}, w_0, \alpha) = \frac{1}{2}\|\vec{w}\|^2 - \sum_{k=1}^{N} \alpha_k \left[e_i(\vec{w}^T\vec{x}_i + w_0) - 1\right] \tag{4.2}$$

où les coefficients α_k sont les multiplicateurs de Lagrange. Le lagrangien doit être minimisé par rapport à $\vec{w}$ et w_0, et maximisé par rapport aux coefficients α_k. La résolution du prob-

lème montre que l'hyperplan optimal ne dépend que des n_s vecteurs supports $\vec{x}_k$ du problème :

$$\vec{w} = \sum_{k=1}^{n_s} \alpha_k \, e_k \, \vec{x}_k \tag{4.3}$$

En notant $\langle ., . \rangle$ le produit scalaire de deux vecteurs, le système d'équations 4.1 devient :

$$\begin{aligned} f(\vec{y}) &= sign\left(\vec{w}^T \vec{y} + w_0\right) \\ &= sign\left(\sum_{k=1}^{n_s} e_k \, \alpha_k \, \langle \vec{x}_k, \vec{y} \rangle + w_0\right) \end{aligned} \tag{4.4}$$

Ce cadre décrit la méthode de résolution des SVM dans un cas linéairement séparable. Cependant, le cas non-linéairement séparable peut être traité en projetant l'ensemble des points à l'aide d'une fonction noyau K. Dans ce nouvel espace, les données seront alors linéairement séparables. Le noyau RBF (4.5) est souvent utilisé pour projeter l'ensemble des points du problème de départ dans ce nouvel espace car il donne généralement de meilleurs résultats qu'un noyau polynomial.

$$K(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2\sigma^2}\right) \tag{4.5}$$

Les SVM sont initialement conçus uniquement pour résoudre les problèmes de classification binaire. La méthode couramment utilisée dans les problèmes de classification ayant plusieurs classes est la méthode « un-contre-un ». Elle consiste à construire un ensemble de classificateurs permettant de séparer une paire de classes entre elles (pour toutes les paires) et à effectuer un vote majoritaire sur l'ensemble des classificateurs pour prendre la décision.

Les SVM sont des classificateurs purs, qui ne vont pas tenter d'inférer des connaissances mais juste de construire un ensemble de séparations optimales entre les différentes classes et ceci uniquement à partir d'exemples fournis. Par ailleurs, il est difficile d'insérer une connaissance *a priori* et complexe dans ce modèle. Cependant, dans certains cas, l'inférence d'information peut être vue comme un problème de classification. Par exemple, pour la reconnaissance d'activité, Fleury et coll. (2010) ont utilisé un modèle SVM qui prend les données des capteurs environnementaux en entrée et fournit la classe d'une activité en sortie.

4.2.1.2 Les réseaux dynamiques et la propagation d'activation

Un réseau dynamique est un réseau de concepts qui évolue avec le temps. Il s'agit d'une représentation graphique d'un ensemble de concepts liés par des relations. Ces relations sont des chemins qui permettent de propager une activation (Crestani, 1997) provenant d'un nœud voisin. Cette approche connexionniste n'est pas très employée dans le domaine de l'habitat intelligent, pourtant elle permet de représenter de manière assez souple les relations entre événements. Une étude récente (Niessen et coll., 2008) a mis en œuvre cette approche pour la reconnaissance de sons en contexte.

Un réseau dynamique est un graphe où les nœuds représentent des concepts et où les arcs, généralement pondérés et orientés, représentent des relations entre les concepts. Un réseau est dynamique dans le sens où le graphe évolue en fonction de l'arrivée de nouveaux

événements qui vont générer de nouveau nœuds dans le graphe. Ces nœuds vont apporter une information qui va se propager dans le graphe et modifier la connaissance courante. On dit que les nœuds sont « activés ». Un nœud activé va propager son activation aux nœud voisins.

L'activation d'entrée A^j_{in} d'un nœud j se calcule généralement par la formule 4.6.

$$A^j_{in} = \sum_{i \neq j} A^i_{out} w_{i,j} \tag{4.6}$$

où $w_{i,j}$ est le poids (c.-à-d., le degré d'association) entre le nœud j et i. L'activation de sortie A^j_{out} d'un nœud j peut être calculée à partir d'une fonction d'activation f telle que définie dans l'équation 4.7.

$$A^j_{out} = f(A^j_{in}) \tag{4.7}$$

La fonction d'activation f peut, par exemple, être une fonction de seuillage ou une fonction sigmoïde. Dans ce cas, la propagation d'activation est équivalente à la propagation dans les réseaux de neurones *feed-forward* classiques. La principale différence vient du fait qu'il n'y a pas véritablement de notion stricte de nœud d'entrée et de nœud de sortie car n'importe quel nœud peut potentiellement être activé par une source extérieure.

Dans le cas d'un graphe cyclique, l'activation peut être difficile à contrôler à cause de boucles dans le réseau. Dans ce cas l'activation peut être freinée par des contraintes sur le chemin emprunté par l'activation, ou en utilisant la sémantique des arcs ou des nœuds (p.ex. un nœud représentant un concept très général fortement connecté doit déclencher moins d'activation qu'un terme précis dans une application en RI), ou encore en imposant une distance maximale de propagation en limitant le nombre de relations traversées.

La propagation d'activation est souvent utilisée en Recherche d'Information dans un réseau sémantique ou simplement associatif pour trouver des documents répondant à une requête. Par exemple, dans (Aswath et coll., 2005), un réseau à deux niveaux où les niveaux de nœud représentent successivement des termes et des produits (produits indexés par les termes) est utilisé pour augmenter le nombre de produits associés à une requête. Dans la première étape de l'approche, une *hot phrase* est associée à des produits qui activent des termes qui à leur tour activent d'autres produits. Le nombre de produits associé à la *hot phrase* devient ainsi plus important que dans le cas d'une association simple *hot phrase*-produits (une activation sur les synonymes de la *hot phrase* est aussi effectuée). Les documents trouvés sont ensuite classés par SVM pour être insérés dans la liste de résultats. La propagation d'activation dans un réseau dynamique offre un cadre flexible et intuitif pour représenter des associations entre concepts, qui deviennent particulièrement intéressant lorsque plusieurs sources d'association sont considérées.

Dans le domaine de la reconnaissance en ligne d'événements, Niessen et coll. (2008) proposent une approche similaire pour améliorer la performance des systèmes de reconnaissance du son en environnements complexes. Dans une application réelle, des sons assez semblables peuvent être produits par plusieurs sources différentes selon le contexte. Dans le travail présenté, les informations de contexte sont utilisées pour diminuer l'incertitude dans

l'identification d'un son. La figure 4.1 montre un exemple d'application de cette méthode. Au départ, c'est-à-dire à gauche de la figure, le son son_{t_1} est identifié comme étant des applaudissements avec une certitude de 100%. Il en découle que l'environnement où l'on se trouve est sûrement celui d'un concert ou d'un match. Un nouvel événement sonore son_{t_2} perçu à t_2, a la même probabilité d'être occasionné par le claquement d'une porte que par le rebond d'un ballon, ces 2 hypothèses sont les sources possibles qui ont pu générer le son. Étant donné le contexte courant, l'hypothèse du claquement de porte peut être diminuée et le contexte de match est renforcé. À partir des événements sonores, des hypothèses et des environnements réseau dynamique hiérarchique est généré où la propagation d'activations prend sa source au niveau des sons (observations). Les liens montrent les relations contextuelles ; par exemple, on peut affirmer avec 80% de certitude que le son d'un ballon se produit lors d'un mach sportif. À la fin du processus d'activation, l'hypothèse la plus probable est celle qui reçoit le plus grand support.

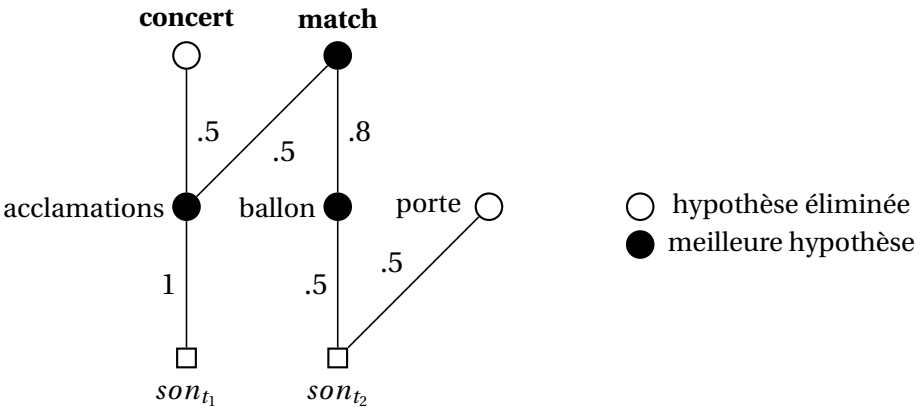


FIGURE 4.1: Réseau dynamique proposé par Niessen pour diminuer l'ambiguïté sonore

La propagation d'activation dans les réseaux dynamiques est une approche très intéressante à mettre en place pour la fusion des données hétérogènes et la résolution de conflits en contexte. Un algorithme de propagation d'activation est relativement facile à développer et à mettre en oeuvre. Par ailleurs, malgré sa puissance d'expression limité et l'absance de méthode adaptée pour son apprentissage automatique, il offre l'avantage de pouvoir prendre en compte l'information temporelle dans la propagation d'activation à travers sa fonction d'activation (Niessen et coll., 2008). Cet avantage a été utilisé dans nos travaux précédents pour le suivi de personnes dans un habitat intelligent à partir de plusieurs sources d'information incertaines. Cette approche est décrite en détail en section 4.3.

4.2.1.3 Modèle de Markov caché

Un modèle de Markov caché ou HMM (*Hidden Markov Model*) (Rabiner, 1989) est caractérisé par un double processus stochastique : un processus interne qui modélise les états non observables $X = \{x_1, x_2 ... x_N\}$ d'un système (qui est sensé être un processus Markovien) et un processus externe qui modélise les observations $Y = \{v_1, v_2 ... v_N\}$ du système dont les

états sont cachés. Les HMM sont très fréquemment utilisés en traitement de la langue naturelle, en bioinformatique et dans tous les domaines où des données sont intrinsèquement organisées en séquences. Cette méthode a donc naturellement été testée dans le domaine de la reconnaissance d'activités en environnements perceptifs en partant du principe qu'une activité humaine est une séquence d'actions atomiques se déroulant les unes à la suite des autres (Wilson et coll., 2005; Singla et coll., 2008; Stikic et coll., 2008; Duong et coll., 2009; Huang et coll., 2010). Dans ce cas, les états du système sont les activités définies et les observations sont les données issues des différents capteurs. On peut aussi citer van Kasteren et Krose (2007) qui utilisent une méthode basée sur un réseau bayésien dynamique (le HMM est une version simple d'un réseau bayésien dynamique) afin de reconnaître 13 activités de la vie courante.

Un modèle HMM noté $\lambda = (A, B, \pi)$ est défini par :

1. L'ensemble de ses états $X = \{x_1, x_2 ... x_N\}$,
2. L'ensemble des observations $Y = \{v_1, v_2 ... v_N\}$,
3. Une matrice A de probabilité de transition entre les états de la chaîne : a_{ij} représente la probabilité de passer de l'état i à l'état j ,
4. Une matrice B de probabilité d'observation : b_{jk} représente la probabilité d'observer le symbole v_k quand le modèle est dans l'état ,
5. Un vecteur π de densité de probabilité initiale, π_i représente la probabilité que l'état de départ du modèle soit l'état i. La distribution de l'état initial $\pi = \{\pi_i\}$.

X prend ses valeurs dans un espace fini discret qui constitue les états du modèle. Y prend ses valeurs dans un espace qui peut être discret ou continu suivant la nature des séquences de données à modéliser. L'observation Y est une fonction probabiliste de l'état. La probabilité d'émission est notée $p(y|x)$, elle est donnée par la matrice B.

Pour illustrer le principe, nous nous cantonnerons au cas d'observations discrètes. La figure 4.2 montre un exemple de chaîne de Markov cachée. À chaque instant t le processus se trouve dans un des états X_i et émet une observation Y_i avec une certaine probabilité ; à l'instant suivant $t+1$, soit il se produit une transition non déterministe vers un autre état, soit le processus reste dans le même état. Dans les 2 cas, une nouvelle observation est générée.

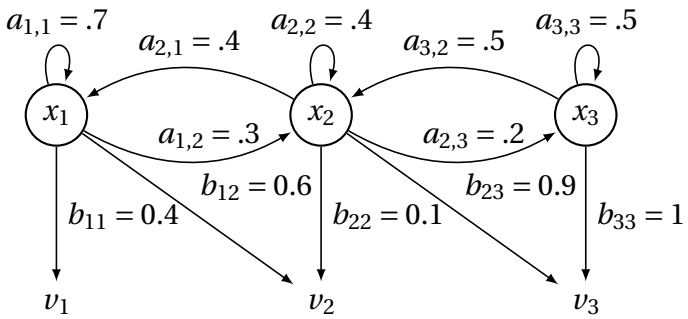


FIGURE 4.2: Exemple de modèle de Markov caché

Les deux tâches d'inférence les plus courantes dans un HMM consistent à : trouver la séquence d'états la plus probable sachant une séquence d'observations, et calculer la probabilité qu'une certaine séquence soit générée.

L'apprentissage des modèles HMM (matrices A, B et probabilités de départ) se fait généralement au moyen de l'algorithme de Baum-Welch qui est une version dérivée de l'algorithme EM (*Expectation-Maximisation*). L'algorithme de décodage le plus utilisé est l'algorithme de Viterbi qui recherche le chemin conduisant à la plus forte probabilité d'émission de la séquence considérée.

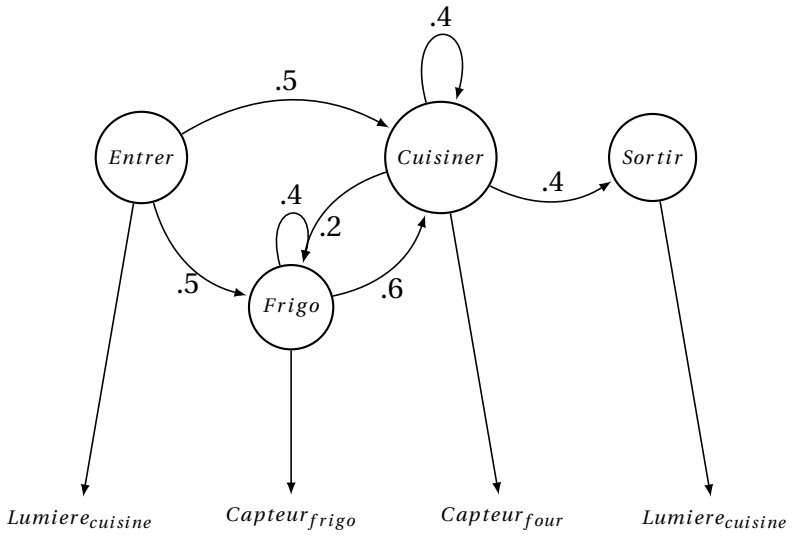


FIGURE 4.3: Modèle de Markov Caché pour l'activité « préparer un repas »

Contrairement aux réseaux bayésien, le HMM est un modèle dynamique étant donné qu'il y a changement d'état en fonction des observations. Grâce à cette propriété, il est plus particulièrement employé dans des domaines où la notion de séquence est importante. Cette approche parait donc bien adaptée au domaine de notre recherche notamment pour reconnaître des activités. La figure 4.3 montre une chaîne de Markov cachée qui modélise la préparation d'un repas. Ici les états sont des actions composant l'activité et les observations sont les informations obtenues des capteurs dans l'appartement. Cependant, ces méthodes nécessitent une taille du corpus d'apprentissage non négligeable et du temps de calcul pour le décodage dans les cas complexes. De plus, il convient de noter que cette approche ne traite naturellement pas des intervalles de temps ni des problèmes d'ordre partiel (c.-à-d. une activité atomique A se déroule au voisinage temporel de B — on ne sait pas si A est avant B ou vice-versa) et de granularité. Ceci conduit à une explosion du nombre de cas si toutes les possibilités doivent être modélisées. Devant cette complexité, bon nombre de recherches en reconnaissance d'activité se sont tournées vers des modèles hiérarchiques. Ces modèles sont l'objet de la section suivante.

4.2.1.4 Modèle de Markov caché hiérarchique

Les HMM peuvent être efficace pour la reconnaissance de certaines séquences d'événements (Wilson et coll., 2005), cependant, ils ne sont pas adaptés à la reconnaissance de séquences dont l'ordre des événements peut varier où lorsque des événements sont manquant. De plus, la notion de durée entre événements n'est pas prise en compte. Pour améliorer ceci, une panoplie d'approches HMM hiérarchiques a vue le jour. Par exemple, Naeem et Bigham (Naeem et Bigham, 2007, 2008) ont utilisé des *Multiple Behavioural Hidden Markov Models (MBHMM)* pour modéliser les différents niveaux d'abstractions d'une tâche de la vie quotidienne et pour permettre de représenter les différentes façons de faire une tâche. Par exemple, la tâche « petit déjeuner » peut être composée de « faire le café » et de « préparer les tartines ». La tâche « faire le café » est elle même composé de séquence d'événements (p.ex. : ouvrir un placard, allumer la machine à café, etc.) qui peuvent être exécutés suivant un ordre différent. Dans l'approche des MBHMM, une hiérarchie d'états cachés représente les niveaux d'activité. Pour chaque activité atomique (c.-à-d. celles inférées à partir des informations du capteur) plusieurs modèles HMM existent pour représenter les différents ordonnancements d'une même sous-tâche (p.ex. : faire le café). Ce modèle, quoique complet, est assez coûteux et ne prend pas en compte la durée.

Le modèle de Markov caché hiérarchique ou HHMM (*Hierarchical Hidden Markov Model*) est lui une généralisation récursive du modèle de Markov caché (Fine et coll., 1998). Dans le modèle hiérarchique, les états sont organisés en arbres, le nombre de sous états d'un état est variable. Chaque état est caractérisé par la matrice de transition et le vecteur de distribution initial de ses sous états. De cette manière, on forme pour chaque état un HMM composé de ses sous états. Les feuilles s'appellent états de production, ils sont les seuls états qui produisent des observations ; en conséquence, ils sont les seuls à avoir une matrice de probabilité d'observation.

Le processus Markovien commence dans l'état racine qui active, à partir de son vecteur de distribution initiale, un état du deuxième niveau et le processus continue récursivement jusqu'à arriver au niveau des états de production où les observations sont générées. Tous les niveaux ont un état final, de telle manière que, quand cet état est atteint, le contrôle retourne à l'état supérieur ; il se produit alors une transition verticale vers un état de même niveau. La figure 4.4 montre un HHMM à quatre niveaux.

Dans (Duong et coll., 2009), une approche combinant les HHMM et les *Hidden Semi-Markov Model (HSMM)* (Yu, 2010) est proposée pour prendre en compte à la fois la structure hiérarchique des activités de la vie courante et la complexité temporelle entre les événements atomiques composant ces activités (la durée est modélisée par une distribution de Cox). L'environnement considéré est une cuisine divisée en 28 carrés, de telle façon qu'un système de suivi puisse localiser l'habitant avec la précision d'un carré élémentaire. Les points de localisations constituent les observations du HHMM lesquels sont ensuite utilisées pour inférer l'activité en cours. Les résultats montrent que le HHMM donne une prédiction légèrement meilleure que les HMM.

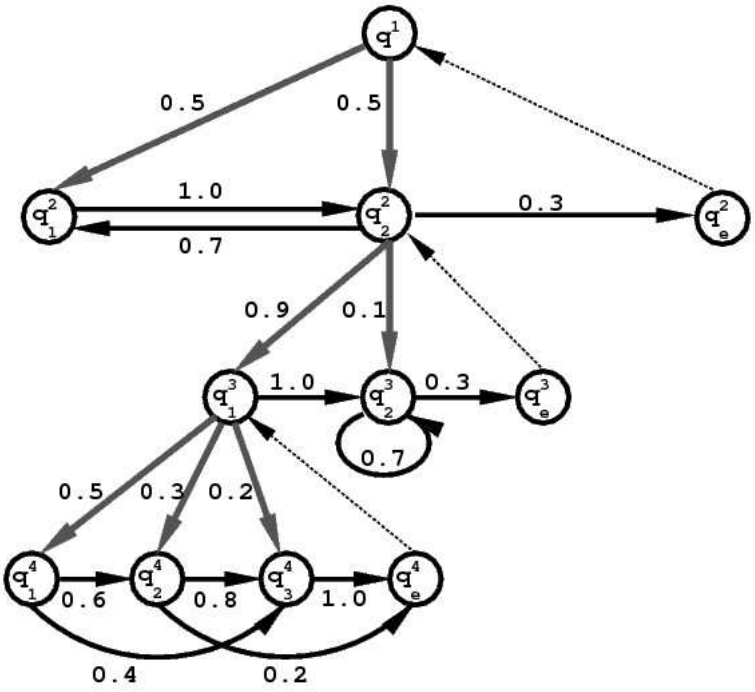


FIGURE 4.4: Exemple de modèle de Markov caché hiérarchique à quatre niveaux

Afin de clarifier ces approches qui se basent toutes sur les HHMM (Tapia et coll., 2004; Duong et coll., 2009; Yu, 2010), il est nécessaire de prendre un exemple concret en rapport avec la reconnaissance d'activité. La figure 4.5 montre un exemple de la modélisation de l'activité « préparer un repas ». Les capteurs placés sur les différents équipements sont notés CP, le capteur du frigo sera donc CP_{frigo}. Ici l'activité principale est décomposée en un premier niveau de sous-activités qui correspond à « préparer une boisson » et à « préparer un plat ». Ces sous-activités sont elles-mêmes composées d'actions qui peuvent être reconnues par les capteurs installés : « prendre un ingrédient dans le frigo », « faire chauffer de l'eau », « chercher quelque chose dans le placard », « se servir du four », « se servir de la cuisinière ».

L'un des problèmes rencontrés lors de l'utilisation des HMM est qu'ils dépendent fortement d'un corpus d'apprentissage qui doit être suffisamment important pour apprendre les paramètres probabilistes. Ceci est décuplé avec les HHMM et les HSMM ou le nombre de paramètres devient encore plus important. L'apprentissage de la structure du modèle est plus complexe dans le cas de ces derniers que dans celui des HMM. Cependant, les méthodes basées sur les HHMM ont montré qu'elles sont bien adaptées aux données d'un habitat intelligent et sont donc des candidates sérieuses pour inférer les événements complexes.

4.2.1.5 Champs aléatoires de Markov

Les champs aléatoires de Markov, aussi appelés réseaux de Markov (RM) (Kindermann, 1980) sont un modèle probabiliste de représentation des connaissances qui se traduit par un graphe non dirigé $G(V,E)$ représentant la probabilité conjointe d'un ensemble de variables aléatoires. Comme dans le cas de réseaux bayésiens, les nœuds correspondent à des variables aléatoires et des décisions, les arcs ont des influences de nature causale ou probabiliste.

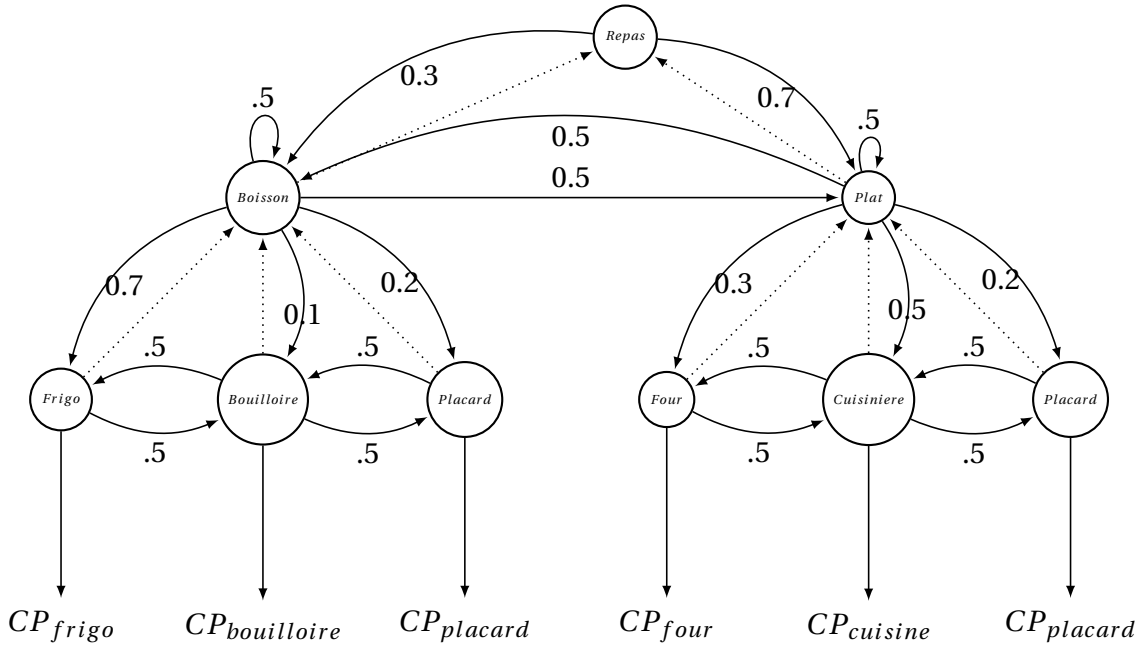


FIGURE 4.5: Modèle de Markov Caché Hiérarchique pour l'activité « préparer un repas »

On fait l'hypothèse que les connaissances disponibles permettent de définir sur l'ensemble des situations possibles une mesure de probabilité unique. Dans le cas d'un processus de Markov, l'état futur $n+1$ dépend uniquement de l'état présent n. Pourtant on peut définir une propriété plus symétrique qui veut que l'état présent n dépend non seulement de l'état précédent $n-1$ mais aussi de l'état postérieur $n+1$. Les réseaux de Markov satisfont à cette propriété de symétrie. Alors que dans les modèles bayésiens, une variable A est influencée par une autre variable B si il y a un arc allant de B vers A, au contraire, dans le cas des RM, toute variable est influencée par les voisines auxquelles elle est liée, ce qui autorise les dépendances cycliques.

Un RM autorise la représentation explicite de dépendance aussi bien que d'indépendance. Dans un modèle représenté par un graphe indirect, si deux variables ne sont pas connectées étant données toutes les autres variables, elles sont indépendantes. La paramétrisation d'un champ de Markov aléatoire est donc différente de celle d'un réseau bayésien. La probabilité conjointe est paramétrée au travers de **fonctions locales** qui n'ont pas une interprétation probabiliste directe mais représentent une contrainte sur les variables aléatoires pour lesquelles la fonction a été définie. Ainsi, pour que des variables indépendantes ne partagent pas la même fonction locale, chaque fonction locale ϕ_k doit opérer sur une clique k, c'est-à-dire un sous-ensemble de nœuds où tous les éléments sont connectés deux à deux.
La distribution conjointe du RM est donnée par :

$$P(X=x) = \frac{1}{Z}\prod_k \phi_k(x_{\{k\}}) \text{ sachant que } Z = \sum_{x\in X}\prod_k \phi_k(x_{\{k\}}) \tag{4.8}$$

où Z est un facteur de normalisation et $x_{\{k\}}$ est l'état de la clique k, c'est-à-dire la configurations des variables qui appartiennent à la clique k.

Un champs aléatoire de Markov doit vérifier la propriété suivante où interviennent les fonctions caractéristiques f :

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_j w_j \cdot f_j(x)\right) \tag{4.9}$$

Les valeurs w_j de l'équation 4.9 sont le résultat d'un problème d'optimisation qui a pour but de trouver les valeurs qui maximisent l'entropie de la probabilité conjointe. L'entropie est une mesure de l'uniformité de l'incertitude parmi toutes les configurations possibles des variables X. Suivant ce principe, les configurations accomplissant les fonctions de caractéristiques reçoivent une probabilité majeure, le reste de la masse probabiliste est réparti uniformément parmi toutes les autres configurations.

Une fonction caractéristique est une fonction binaire. Un exemple de fonction caractéristique appliquée à un ensemble de trois variables peut être définie en s'inspirant du cas suivant :

$$f(x_1, x_2, x_3) = \begin{cases} 1, & \text{si } x_1 < 0.5 \text{ et si } x_2 = x_3 \\ 0, & \text{sinon} \end{cases} \tag{4.10}$$

Une configuration des variables pour laquelle la fonction caractéristique f a une valeur de 1 aura une probabilité plus grande. Ces fonctions de caractéristiques sont apprises à partir des données d'apprentissage et représentent des statistiques importantes à considérer dans le modèle.

La méthode la plus utilisée pour réaliser l'inférence dans un RM est la chaîne de Markov Monte-Carlo (MCMC) (Gilks et coll., 1995).

La principale contrainte de l'application de cette approche pour l'inférence d'information dans notre recherche est la difficulté de modéliser des relations temporelles. D'ailleurs, étant un modèle non-dirigé, la conception de la structure peut être très complexe puisque il ne suffit pas d'analyser les dépendances cause-effet entre les variables (comme dans les cas des méthodes basés sur des graphes dirigés), mais il faut considérer des autres types de relations qui pourraient donner lieu à des dépendances cycliques. Les méthodes d'apprentissage automatique de structure ne sont pas très optimales. On ne trouve pas d'applications utilisant directement cette méthode dans l'état de l'art de l'inférence des informations contextuelles dans les environnements perceptifs.

4.2.2 Approches logiques

Dans cette catégorie, les méthodes réalisent le raisonnement en appliquant les principes de l'inférence logique (déduction, abduction ou induction). Leurs principales avantages sont leur puissance d'expression, leur modèle exprimés de manière déclarative et le fait que leur satisfaisabilité peut être évaluée Artikis et coll. (pear). Avant de décrire quelques méthodes dans ce domaine, il est nécessaire de définir les concepts les plus importants de la logique

en intelligence artificielle.

4.2.2.1 Concepts de la logique pour l'intelligence artificielle

Les différents concepts utilisés peuvent être definis de la façon suivante :

- **Une constante** a**,** représente un objet d'un domaine d'intérêt (p. ex., `lundi, pierre, cuisine`).
- **Une variable** X**,** prend une valeur du domaine d'intérêt. Une instanciation est l'assignation d'une valeur (constante) à une variable, $X = a$.
- **Un terme** t est une variable ou une constante. Un terme instancié est une constante.
- **Un prédicat** P est une fonction booléenne ayant un ensemble $T = \{t_1, t_2, .., t_n\}$ de termes en paramètre. Donc $P : T \rightarrow \{vrai, faux\}$. Les prédicats peuvent représenter un attribut d'un objet (p. ex. Petit(X)), ou une relation entre objets(p. ex., PereDe(X,Y)). Un **prédicat instancié** a comme paramètres uniquement des termes instanciés.
- **Une fonction logique** f représente une correspondance d'un ensemble de termes T_1 vers un autre ensemble T_2. $f : T_1 \rightarrow T_2$ (p. ex., EnfantsDe(X,Y)).
- **Un atome** est un prédicat avec ses paramètres.
- **Un littérale** l**,** est un atome ou la négation d'un atome ($\neg$).
- **Une formule logique** F**,** est un atome, ou une construction qui utilise des atomes, connecteurs logiques ($\wedge, \vee$), et quantificateurs ($\forall, \exists$). Ainsi, si F_1 et F_2 sont des formules, $\neg F_1$ et $F_1 \wedge F_2$.
- **Une base de règles** est un ensemble de formules logiques.
- **Une clause C** est un littérale ou une disjonction de littéraux ($l_1 \vee ... \vee l_n$). Il faut noter qu'une implication peut s'exprimer comme une clause, $l_1 \Rightarrow l_2 \equiv \neg l_1 \vee l_2$.
- **Une clause de Horn** est une clause comportant au plus un littéral positif.
- **Un programme logique** PL**,** est un ensemble de clauses logiques.
- **Un alphabet** Σ est l'ensemble des prédicats, symboles, constantes, et fonctions pour décrire logiquement un domaine.
- **Une base de Herbrand** d'un programme logique PL, $HB(PL)$, est l'ensemble de tous les atomes instanciés (avec tout les combinaisons possibles d'assignation à ses variables) dans l'alphabet de PL.
- **Une interprétation** I**,** est l'assignation d'une signification sémantique aux éléments d'un alphabet. Par exemple, l'interprétation de l'atome PereDe(X,Y) est : X et Y sont deux personnes et X est le père de Y.
- **Un monde possible ou interprétation de Herbrand** IH**,** est une assignation de vérité à tous les prédicats instanciés d'un domaine. Il est possible de considérer IH comme le sous ensemble de la base de Herbrand où les prédicats sont vrais.

4.2.2.2 Systèmes d'inférence à base de règles logiques

Les systèmes à base de règles ont l'avantage d'être très simple à mettre en œuvre étant donné leur analogie avec le raisonnement humain (logique formelle) et le nombre de sys-

tèmes existants (Prolog, CLIPS, Jess, etc.). La plupart des moteurs d'inférence reposent sur des mécanismes de logique formelle. L'activation des règles se fait généralement par chaînage avant, arrière ou mixte. Les règles sont de la forme '**SI** *condition* **ALORS** *action*', elles sont en général définies par un expert du domaine. Le moteur d'inférence génère des conclusions en faisant une évaluation des règles.

Pour la reconnaissance dans le domaine des habitats intelligents, quelques travaux ont utilisé des approches fondées sur des règles logiques. Storf et coll. (2009) décrivent une application multi-agents qui détermine les activités de l'habitant et évalue si le contexte est normal ou dangereux. Les règles sont insérées dans la base de connaissance du système par l'expert via une interface. Elles sont stockées comme un ensemble d'événements qui caractérisent une activité de la vie quotidienne (AVQ) ainsi que les relations temporelles associées qui permettent de préciser la séquence d'événements qui composent une AVQ. Chaque activité est décrite de différentes manières puisque dans la réalité une même activité peut être générée par plusieurs séquences d'événements. L'algorithme d'inférence prend comme entrée une liste de faits et la compare avec les séquences d'événements définissant les activités, puis il en déduit un poids de similarité pour chaque AVQ définie dans la base de règle et sélectionne la plus similaire. Une approche ayant pour base l'apprentissage de règles d'association a ensuite été définie pour acquérir les règles automatiquement.

Une solution similaire est décrite par Augusto et Nugent (2004). Dans ce travail, les règles modélisent l'aspect temporel des événements pour la reconnaissance de situations par l'inclusion des intervalles de temps comme paramètres des prédicats logiques qui représentent les événements. L'exemple suivant décrit une règle que sera déclenchée si l'habitant quitte la cuisine en laissant le four allumé pendant plus de 10 minutes.

$$ON(occurs(start(oven), I1a][I1b)\text{^}occurs(transition(kitchen, bedroom), I2a][I2b))$$
$$IF(earlier(I1b, I2b)\text{^}\neg holds(at_kitchen, [I2b, Now])\text{^}moreThan(I2b, Now, 10\ mins))$$
$$THEN(ApplyHazardProcedure\text{^}TryContact)$$

L'approche par modèles logiques est une approche appropriée pour modéliser les situations parce qu'elle permet de représenter la connaissance d'une façon assez expressive et compréhensible avec une grande capacité de généralisation, lorsque par exemple, la logique employée est du premier ordre. De plus, l'approche étant déclarative, la mise en œuvre de l'algorithmique d'inférence est transparente pour l'utilisateur. Cependant, ces approches peuvent difficilement traiter l'incertitude car celle-ci ne se modélise pas naturellement avec des règles logiques.

4.2.2.3 Programmation logique inductive (ILP)

La programmation logique inductive (*Inductive Logic Programming — ILP*) est la combinaison de l'apprentissage automatique et la programmation logique (Muggleton et Raedt, 1994). L'objectif de cette méthode est d'induire des théories à partir d'exemples sous forme de programmes logiques. Elle tire de l'apprentissage automatique le principe d'induction des hypothèses à partir des données d'apprentissage, mais sa représentation basée sur la

logique de premier ordre permet l'induction de théories plus expressives que les approches d'apprentissage automatique classiques qui induisent des hypothèses propositionnelles. De plus, la connaissance *a priori*, celles d'un expert, peut être facilement utilisée pour guider l'apprentissage.

Les éléments logiques impliqués dans l'ILP peuvent être définis ainsi :

- un ensemble d'exemples positifs E^+ et un ensemble d'exemples négatifs E^-. Ce sont typiquement des variables littérales instanciés ;
- un langage d'hypothèse $\mathscr{L}_H$, à partir duquel les hypothèses H sont crées ;
- une base de connaissances B. B et H sont des ensembles de clauses de la forme $\alpha \leftarrow l_1, \ldots, l_n$.

Le problème d'ILP est de chercher les hypothèses $H \subseteq \mathscr{L}_H$, telles que $B \wedge H \models E^+$ (propriété de complétude) et $B \wedge H \wedge E^- \not\models \square$ (propriété de cohérence) (Muggleton et Raedt, 1994). La condition de complétude garantit que tous les exemples positifs de E^+ peuvent être déduits de H et B. La condition de cohérence garantit qu'aucun exemple négatif de E^- ne peut être déduit de H et B.

Considérons par exemple la découverte des relations existantes entre les personnes d'une famille. Imaginons que l'on sache que le grand-père c'est le père d'un parent, mais que l'on ne connaisse pas la définition d'un parent. Nous disposons de la base de connaissances suivante :

$$
\begin{aligned}
B = \{ & grandPere(X,Y) \leftarrow pere(X,Z), parent(Z,Y) \\
& pere(henry, jane) \\
& mere(jane, john) \\
& mere(jane, alice)\}
\end{aligned}
$$

De plus, les exemples suivants sont supposés connus :

$$
\begin{aligned}
E^+ = \{ & grandPere(henry, john) \\
& grandPere(henry, alice)\} \\
E^- = \{ & grandPere(john, henry) \\
& grandPere(alice, john)\}
\end{aligned}
$$

Par l'application de l'ILP, nous pouvons arriver à l'hypothèse suivante qui permet de déduire E^+ mais aucun E^- :
$H = parent(X,Y) \leftarrow mere(X,Y)$
L'avantage le plus évident de l'ILP est de permettre de trouver des règles générales qui expliquent certains problèmes à partir de règles de base et quelques exemples. Nous n'avons pas trouvé d'applications directes d'ILP pour l'inférence d'informations dans l'intelligence ambiante ; pourtant, cette méthode pourrait être appliquée pour trouver des règles logiques qui servent à enrichir une base de connaissances. En outre, il est possible d'utiliser des données expérimentales pour apprendre des règles qui servent à créer un modèle logique d'inférence des informations. Ainsi, bien qu'ILP ne soit pas appliqué directement dans des problèmes d'inférence, c'est un outil pour l'apprentissage de la structure des autres modèles.

4.2.2.4 Logique floue

La logique floue, introduite par Zadeh (1965), est une branche de l'Intelligence Artificielle qui traite de la représentation et du raisonnement à partir de connaissances imprécises. En effet, dans de nombreux cas la connaissance des experts est subjective et approximative. À la différence de la logique booléenne qui ne considère que deux états exclusifs pour une proposition logique, la logique floue peut utiliser un ensemble d'états (voire une distribution continue) dans lequel une proposition prend sa valeur de vérité. Les systèmes d'inférence floue (SIF) sont capables de raisonner à partir de cette connaissance imprécise et de déduire des réponses parfois précises. Par exemple, la logique floue est souvent employée en automatique comme dans le cas de la régulation du chauffage (Gacôgne, 1997).

La logique floue a été appliquée dans plusieurs projets de maisons intelligentes notamment EMERGE (Storf et coll., 2009), CompanionAble (Medjahed et coll., 2009) ou AOFIS (Adaptive Online Fuzzy Inference System) (Doctor et coll., 2005). Dans ce dernier, les données de plusieurs capteurs sont utilisées pour activer des règles de logique floue qui ont été créées pour représenter le comportement de l'habitant. Le but du projet AOFIS est la création d'un système qui puisse générer ces règles pendant un processus d'apprentissage avant de les utiliser pour contrôler des actionneurs agissant sur la lumière ou le chauffage de façon à améliorer le confort de l'habitant.

Les SIF représentent l'imprécision au travers d'*ensembles flous*. Ces ensembles modélisent le degré d'appartenance d'une valeur à une ou plusieurs propriétés. Étant donnée une variable $x \in U$, avec U le domaine de valeur fini ou non de x et une classe représentée par un ensemble flou F, nous définissons une fonction d'appartenance $\mu_F(x)$ prenant ses valeurs dans l'intervalle $[0,1]$. Cette fonction indique le degré d'appartenance de la variable x à la classe représentée par $F = \{(x, \mu_F(x)) \mid x \in U\}$. La figure 4.6 montre l'appartenance de la température (variable x) d'un habitat aux ensembles floues « froid » ou « chaud ». Nous pouvons noter dans cet exemple que pour une température de 30°C, x appartient avec un degré de 0,25 à « froid » et avec un degré de 0,75 à « chaud » ; la température n'appartient pas exclusivement à un de ces ensembles, et c'est ainsi qu'on modélise l'imprécision.

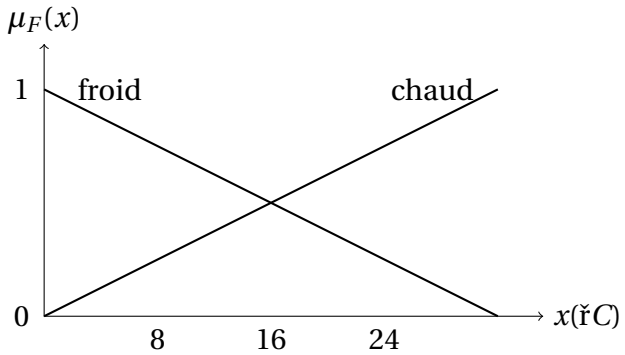


FIGURE 4.6: Fonctions d'appartenance selon la température d'un habitat

En logique floue les opérations d'égalité, d'inclusion, d'union, d'intersection, et de com-

plément sont définies en utilisant la fonction d'appartenance, par exemple nous dirons que deux ensembles flous A et B sont égaux dans un univers U si leurs fonctions d'appartenance ont la même valeur en tout point $x \in U$:

$$\mu_A(x) = \mu_B(x)\,, \forall x \in U\,; \text{ en conséquence, si } \exists x \text{ tel que } \mu_A(x) \neq \mu_B(x) \Rightarrow A \neq B$$

La fonction d'appartenance est applicable également aux relations entre éléments de deux ensembles flous pour représenter le degré d'association ou d'interaction. Des exemples de ces relations sont : x est plus grand que y, ou x est moins froid que y. Les relations floues sont importantes dans la conception d'applications réelles. Formellement, une relation floue $R(U_1, U_2)$ définie dans les univers U_1 et U_2 est un ensemble flou dans l'espace U, U étant défini comme le produit cartésien $U_1 \times U_2$. Cette relation floue est caractérisée par sa fonction d'appartenance $\mu_R(x, y)$ où $x \in U_1$ et $y \in U_2$, $R(U_1, U_2) = \{((x, y), \mu_R(x, y)) \mid (x, y) \in U = U_1 \times U_2\}$

La logique floue définie des opérateurs pour pouvoir réaliser des opérations logiques. Il convient de noter que les propositions $A \cup \bar{A} \neq U$ et $A \cap \bar{A} \neq \emptyset$ peuvent être vraies et c'est pour cette raison les opérations mathématiques concernant les ensembles flous ne sont pas définies de la même façon que pour les ensembles classiques.

$$\begin{cases} \mu_{A\cup B}(x) = max[\mu_A(x), \mu_B(x)] \\ \mu_{A\cap B}(x) = min[\mu_A(x), \mu_B(x)] \\ \mu_{\bar{A}}(x) = 1 - \mu_A(x) \end{cases} \tag{4.11}$$

La figure 4.7 donne le schéma général d'un système d'inférence floue (SIF). Un SIF est composé d'un étage de *fuzzification*, d'un moteur d'inférence qui utilise les règles floues et d'un étage de *defuzzification*. Succinctement, les trois étapes classiques de l'inférence floue consistent à 1) décrire des variables d'entrée (p.ex. la vitesse est de 65km/h) dans des ensembles catégoriques plus vagues (p.ex., le véhicule avance rapidement) — c'est l'étape de fuzzyfication ; 2) activer des règles expertes floues (p.ex., si la vitesse est grande il faut la réduire légèrement) — c'est l'étape d'inférence ; et 3) les valeurs floues sont retranscrites dans un autre domaine qui peut être numériques (p.ex., baisser la vitesse de 15 km/h) — c'est l'étape de *defuzzification*.

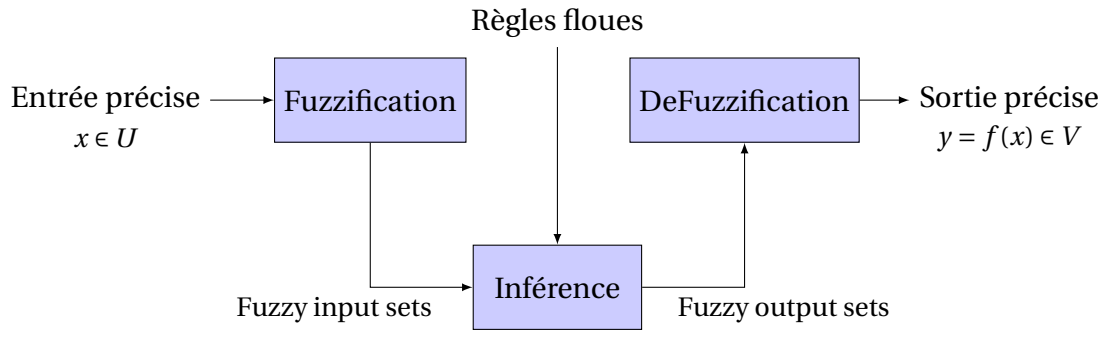


FIGURE 4.7: Un système d'inférence floue général (Mendel, 1995)

Pour décrire l'intérêt de la logique floue dans le cadre de l'inférence en contexte, il est pertinent de la comparer avec la théorie probabiliste. La différence entre ces deux approches résident dans la signification de ce qu'elles modélisent. Un modèle probabiliste cherche à mesurer la vraisemblance de l'occurrence d'un événement, tandis que la logique floue calcule une valeur de vérité pour un fait. Si l'on se place dans le cadre de la reconnaissance d'activité, en utilisant des méthodes probabilistes nous pourrions obtenir la probabilité que l'activité courante soit dormir étant donné les observations courantes. Dans ce cas, l'incertitude est due au fait que l'information que nous avons n'est pas suffisante pour affirmer avec certitude quelle est l'activité en cours. Si on connaît la vérité terrain, cette incertitude n'existe plus. Par contre, un SIF considère que toute l'information est disponible. Cette approche pourrait nous donner une valeur de vérité par rapport à l'appartenance de l'activité courante à la classe dormir, mais comme nous considérons que l'habitant fait toujours une seule activité à la fois cette valeur serait toujours 1. Si on connaît la vérité terrain, le dégrée de vérité est toujours le même. Ainsi, la logique floue permet de raisonner sur des données imprécises mais non incertaines (il est certain que 16 n'*est pas très chaud*) alors que l'approche probabiliste permet de raisonner sur des données incertaines afin d'obtenir l'hypothèse la plus vraisemblable.

De toute évidence, pour l'inférence des informations contextuelles, la théorie de probabilité est celle qui a été la plus employée. Ce choix est notamment dû au fait que le problème est d'inférer des événements dans un environnement où l'information n'est pas disponible dans sa totalité. La logique floue est, par contre, une approche très intéressante pour raisonner avec des critères vagues tels que le niveau de sécurité ou de confort pour qualifier la situation courante.

4.2.2.5 Raisonnement par ontologies

Une ontologie est définie comme une spécification explicite d'une conceptualisation (Gruber, 1993). Les ontologies représentent formellement des connaissances comme un ensemble de concepts et leur relation dans un domaine. Elles connaissent un essor considérable avec le développement du web sémantique. D'une manière générale les ontologies sont souvent utilisées dans les applications informatiques pour définir formellement les données échangées afin de facilité la transmission des données entre services et de permettre des requêtes de haut niveau. Son développement au sein des systèmes intelligents est dû non seulement à sa capacité de représentation mais également au support qu'elle offre aux fonctions de communication, réutilisation et raisonnement. Dans notre recherche, nous nous intéressons aux ontologies formalisées par le *Ontology Web Language* (OWL) (Dean et Schreiber, 2004), qui définit un langage de spécification des ontologies qui se base sur la logique de description (*Description Logic — DL*). La logique de description (Baader et coll., 2008) est un type de logique permettant de décrire des concepts et leurs rôles (relations). Elle se base sur la logique du premier ordre avec l'idée de formaliser des réseaux sémantiques. Cependant, son langage s'écarte de la logique du premier ordre. En effet la principale motivation est d'obtenir un langage suffisant pour les tâches de modélisation et pour

avoir de bonnes propriétés de décidabilité, de consistance et de satisfaisabilité. La principale différence entre la logique de description et la programmation logique (p.ex. Prolog) est l'*hypothèse du monde ouvert* selon laquelle on ne peut pas affirmer ou nier la vérité d'une clause si elle n'est pas définie explicitement ou si elle ne peut pas être déduite directement de la connaissance présente. Dans la programmation logique, par contre, on utilise la *négation par l'échec* : s'il n'est pas possible prouver la vérité d'une clause, on la considère fausse.

Un des objectifs principaux de OWL est de faciliter l'intégration de capacités de raisonnement en logiques de description dans le web sémantique. Trois types de OWL ont été définis. *OWL Lite* est le plus élémentaire des langages OWL et peut être utilisé pour exprimer des taxonomies et restrictions simples. *OWL DL* supporte plus d'expressivité tout en assurant la complétude et la décidabilité pour le raisonnement. *OWL Full* n'a aucune restriction d'expressivité, par contre la complétude et la décidabilité ne sont pas assurées.

La représentation des connaissances basées sur la logique de description comporte deux composantes : la **TBox** et la **ABox**. La TBox décrit la terminologie, les définitions des concepts et des relations. La ABox contient des individus qui sont vus comme des instances de concepts. Les concepts sont vus comme des ensembles regroupant des individus et les rôles décrivent des relations entre les individus. Le tableau 4.1 donne les éléments de base de la logique de description ainsi que leur sémantique. Dans ce tableau, A est un concept atomique, C et D sont des descriptions de concepts (concepts atomiques ou composés), et R est un rôle. La sémantique est définie en utilisant une interprétation, I, qui est constituée de deux éléments : 1) un ensemble Δ^I (le domaine d'interprétation qui regroupe tous les individus) et 2) une fonction d'interprétation qui assigne un ensemble $A^I \subseteq \Delta^I$ à tout concept atomique A, et une relation binaire $R^I \subseteq Delta^I \times \Delta^I$ à tout rôle atomique R.

Syntaxe	Sémantique	Description
A	$A^I \subseteq \Delta^I$	Concept atomique
R	$R^I \subseteq \Delta^I \times \Delta^I$	Rôle atomique
$\top$	Δ^I	concept subsumant tous les autres (plus général)
$\bot$	$\emptyset$	concept vide (plus spécifique)
$\neg A$	$\Delta^I \backslash A^I$	Négation atomique
$C \sqcap D$	$C^I \sqcap D^I$	Intersection
$C \sqcup D$	$C^I \sqcup D^I$	Union
$C \sqsupseteq D$	$C^I \sqsupseteq D^I$	Subsomption
$\forall R.C$	$a \in \Delta^I \mid \forall b.(a,b) \in R^I \Rightarrow b \in C^I$	Restriction de valeur
$\exists R.C$	$a \in \Delta^I \mid \exists b.(a,b) \in R^I \wedge b \in C^I$	Quantificateur existentiel
$\geq n\ R$	$a \in \Delta^I \mid \mid b \mid (a,b) \in R^I \mid \geq n$	Restriction de nombre
$\leq n\ R$	$a \in \Delta^I \mid \mid b \mid (a,b) \in R^I \mid \leq n$	

TABLE 4.1: Syntaxe et sémantique de la logique de description

Le raisonnement consiste à dériver des faits qui n'ont pas été explicitement exprimés dans l'ontologie. Les tâches de raisonnement pour une ontologie sont principalement les suivantes :

– **Satisfaisabilité.** Déterminer si la définition d'un concept n'est pas contradictoire, c'est à dire s'il peut y avoir un individu qui soit une instance du concept.

- **Subsomption.** Déterminer si un concept C subsume un concept D.
- **Consistance.** Déterminer si les individus ne violent pas les descriptions et les axiomes qui décrivent les concepts.
- **Équivalence.** Si la définition d'un concept C est égale à celle d'un concept D.
- **Disjonction.** Si l'ensemble d'individus d'un concept C est disjoint de celui du concept D.

La définition d'une activité de vie quotidienne, « préparer le repas », sera la suivante en utilisant la logique de description :

$$\begin{aligned} SortirVaisselle &\sqsubseteq DureeCourte \sqcap \exists Utilise.(Objet \sqcap \forall SitueDans.Furniture(placard)) \sqcap \forall Produce.Son(vaisselle) \\ PreparerRepas &\sqsubseteq DureeLongue \sqcap \exists Inclue.SortirVaisselle \sqcap \forall SitueDans.Localisation(cuisine) \end{aligned} \tag{4.12}$$

Dans cet exemple, les instances des concepts $SortirVaisselle$ et $PreparerRepas$ peuvent être des séquences d'événements ayant une heure de début et fin spécifique. Selon cette définition, l'action de sortir la vaisselle est mise en évidence par l'utilisation des objets qui sont placés dans le placard, et lors de l'action il est possible d'entendre des sons de vaisselle. Si cette action est faite dans la cuisine, l'habitant peut être en train de préparer son repas.

La création d'une ontologie implique la compréhension du domaine de connaissance. Ce processus est souvent mis en œuvre avec l'aide d'un expert du domaine. Deux facteurs sont à considérer lors de la création d'une ontologie : l'objectif final (représentation, raisonnement, les applications qui vont exploiter les connaissances) et l'ampleur de la couverture du domaine afin d'inclure uniquement les éléments pertinents. Dans l'état de l'art des systèmes d'intelligence ambiante, toutes les ontologies utilisées dans les applications ont été construites par expertise. Pourtant, il existe des méthodes pour l'apprentissage automatique ou semi-automatique d'ontologies (Hazman et coll., 2011). Cependant, la plupart de ces méthodes permettent d'extraire des ontologies à partir de textes.

Comme pour les systèmes à base de règles logiques, le langage de description des ontologies présente une grande expressivité. Même si les capacités de raisonnement sont plus limitées que ceux des systèmes de programmation logique, l'avantage des interfaces de gestion des ontologies est d'inclure non seulement un mécanisme d'inférence de nouvelles connaissances mais aussi une plate-forme de représentation qui permet la réutilisation et le partage des connaissances entre les différents composants d'un système. Mais comme pour les systèmes logiques, cette description a ses limites et notamment cette approche ne peut traiter l'incertitude. Malgré cela, lorsque les informations de haut niveau ne sont pas affectées par l'incertitude, le raisonnement par les mécanismes embarqués dans les ontologies est très intéressant car il se fait directement dans le modèle de connaissances et la mise à jour des descriptions dans l'ontologie est faite automatiquement. Le couplage entre la connaissance déclarative et le raisonnement est très fort.

4.2.2.6 SWRL

Semantic Web Rule Language (SWRL) (Horrocks et coll., 2004) est un langage de règles proposé par le W3C qui combine OWL-DL avec RuleML (Rule Markup Language), un langage de balisage pour la mise en oeuvre de règles logiques. Les règles en SWRL sont spécifiées à travers des clauses de Horn dont les atomes sont des concepts, relations, ou fonctions définis dans une ontologie. Ainsi, les atomes ont une des formes suivantes : $C(x)$, $P(x,y)$, $sameAs(x,y)$, $differentFrom(x,y)$, où x et y sont des variables ou des individus, P est une relation (rôle) OWL, et C est la description d'une classe (concept). Il est également possible d'utiliser des atomes *Build-in*, des prédicats qui permettent d'étendre la fonctionnalité des règles par l'inclusion d'opérations arithmétiques, de comparaison, de traitement des caractères ou de dates.

L'un des avantages de SWRL est d'étendre l'expressivité de OWL-DL. Par exemple, en OWL-DL il n'est pas possible de définir une relation comme étant la composition de deux autres relations. Un exemple d'une telle définition en utilisant SWRL est la suivante :

$$parent(?b,?a) \wedge frere(?b,?c) \rightarrow oncle(?c,?a)$$

La mise en œuvre de bases de connaissances par des ontologies a été très appliquée récemment dans les systèmes de contrôle de maisons intelligentes. Il existe notamment plusieurs travaux utilisant SWRL pour la reconnaissance de situations en contexte (Ricquebourg et coll., 2007; Ruan et coll., 2010). Par exemple, Ricquebourg et coll. (2007) utilise l'inférence par SWRL pour contrôler la domotique. L'exemple de règle SWRL suivant est utilisé pour allumer le radiateur quand la température tombe au dessous d'une valeur limite :

$$Radiateur(?r) \wedge Temperature_Sensor(?t) \wedge Room(?ro) \wedge Occupied_house(?h) \wedge IsLocated(?r,?ro) \wedge IsLocated(?t,?ro) \wedge IsLocated(?r,?h) \wedge LowValue(?r,?trad) \wedge SensingValue(?t,?tcapt) \wedge swrlb:lessThan(?tcapt,?trad) \rightarrow Radiator_OFF(?r)$$

Si SWRL permet d'augmenter l'expressivité de OWL, il peut, par contre, rendre l'inférence dans une ontologie indécidable (Parsia et coll., 2005), c'est-à-dire qu'il n'est pas possible de calculer en temps fini si un axiome est impliqué par la base de connaissances ou pas. Dans la plupart des cas, l'utilisation de OWL et SWRL a pour seul but la représentation de connaissances et l'indécidabilité imposée par SWRL n'est pas un frein à son utilisation. Par contre, si l'on cherche à utiliser un raisonneur pour inférer des connaissances dans la base de connaissances il existe des restrictions à mettre en place dans les règles pour que le modèle reste décidable. La plupart des outil implémentant SWRL forcent la décidabilité en restreignant les règles à être *DL-safe rules* (Kolovski et coll., 2006). Une règle est *DL-safe* si toutes ses variables sont attachés seulement à des individus explicitement mentionnés dans l'ABOX.

L'inférence des informations par SWRL a les mêmes inconvénients que celle employant OWL-DL : l'impossibilité de traiter l'incertitude des sources d'information. Cependant, cette méthode offre la capacité d'exprimer des relations complexes en utilisant des éléments qui ont été préalablement définis dans l'ontologie, une caractéristique très intéressant pour la reconnaissance de situations où les descriptions de situations d'intérêt sont complexes et

impliquent souvent des relations spatio-temporelles qui peuvent être représentées par des rôles. Une fois que les situations sont définies, l'inférence consiste à évaluer si la définition correspond à l'état courant dans la base de connaissance.

4.2.3 Approches statistique-relationnelles

L'apprentissage statistique-relationnel (Getoor et Taskar, 2007) est un sous-domaine de l'Intelligence Artificielle et de l'Apprentissage Automatique qui concerne l'induction de modèles qui traitent non seulement de l'incertitude à travers des méthodes statistiques mais aussi de la complexité des structures relationnelles. Dans cette partie, nous décrivons des méthodes représentatives de cette approche qui peuvent être appliquées à l'inférence d'informations contextuelles. Bien d'autres méthodes existent telles que celles décrites par Raedt et coll. (2007), Milch et coll. (2005), etc., que nous ne détaillerons pas dans ce mémoire.

4.2.3.1 Modèle de Markov caché logique

Un modèle de Markov caché logique (ou *Logical Hidden Markov Model* : LoHMM) (Kersting et coll., 2006) traite des séquences de symboles structurées sous forme d'atomes logiques au lieu de propositions. Les HMM sont essentiellement des modèles propositionnels, c'est-à-dire, qu'ils utilisent des symboles qui ne permettent pas de faire des généralisations. La logique du premier ordre va au delà de la logique propositionnelle en généralisant des prédicats qui prennent des variables en paramètres.

Dans cette approche (Kersting et coll., 2006), les transitions dans les LoHMM prennent la forme suivante :

$$p : H \overset{O}{\leftarrow} B \tag{4.13}$$

où p désigne la probabilité d'exécuter la transition, H et B des états abstraits définis par des prédicats logiques, et O le symbole émis.

Considérons l'exemple d'un LoHMM modélisant l'activité « faire le ménage » dans une pièce $Piece$ de l'appartement en utilisant un objet de nettoyage $Objet$ à un instant de la journée $Temps$. Les états caractérisent les événements de l'activité, les observations sont les informations obtenues des capteurs dans les transitions entre états. Il peut donc y avoir la transition de l'exemple suivant entre les 2 états $Entrer$ et $NettoyerSol$:

$$0.8 : NettoyerSol(Piece, Objet) \overset{Presence(Piece)}{\longleftarrow} Entrer(Piece, Temps)$$

Un LoHMM peut être considéré comme un méta-modèle, c'est-à-dire qu'on ne fait pas d'inférence sur ce modèle jusqu'à ce que toutes les variables soient instanciées. L'utilisation des éléments de logique de premier ordre permet de généraliser un problème, mais le

raisonnement est fait sur des instances. Dans les LoHMM, on ne définit pas des matrices de transitions et d'observations comme dans les HMM, le modèle est composé d'un ensemble de transitions logiques du même type que celle de l'exemple. Dans l'exemple, la variable $Piece$ est commune aux deux états ainsi que l'observation $Presence$. Cela permet d'indiquer que l'instance utilisée pour cette variable doit être la même dans tous les cas ; l'information partagée entre les états est alors unique, cette unicité est particulièrement appropriée dans le cas de problèmes utilisant des instances structurées.

L'un des problèmes à résoudre dans les LoHMM est le fait que certaines transitions sont non déterministes. Dans l'exemple, bien que la variable $Piece$ reste invariable lors de la transition, la valeur de la variable $Objet$ reste à déterminer parmi un ensemble d'instances possibles. Par exemple, si on se trouve dans l'état $Entrer(cuisine, apresMidi)$, la règle de transition indique comme état suivant $NettoyerSol(cuisine, Objet)$ sans donner d'indication pour fixer la variable $Objet$.

Pour résoudre ce problème, les opérations d'inférence dans les LoHMM se font par l'intermédiaire d'une fonction de sélection μ qui détermine l'instance qui correspond aux variables d'un prédicat. Cette fonction est probabiliste. Dans l'exemple, si on considère les valeurs de μ suivantes et que l'on est dans l'état $NettoyerSol(Piece, Objet)$:

$$\mu(NettoyerSol(Piece, Aspirateur) \mid NettoyerSol(Piece, Objet)) = 0.6\,;$$

$$\mu(NettoyerSol(Piece, Serpilliere) \mid NettoyerSol(Piece, Objet)) = 0.4\,;$$

la fonction μ permet d'unifier les variables $Piece$ et $Objet$ avec $cuisine$ et $aspirateur$ respectivement. Les valeurs assignées par la fonction μ sont apprises à partir des données d'apprentissage. Dans l'exemple, elles représentent les probabilités d'observer chacune des instances $aspirateur$ et $serpilliere$ quand on est dans l'état $NettoyerSol(Piece, Objet)$.

Il est possible que l'on se trouve dans un cas où plusieurs règles de transition sont applicables en même temps. Considérons l'exemple des deux règles de transition suivantes :

$$0.4 : NettoyerSol(Piece, Objet) \overset{Presence(Piece)}{\longleftarrow} Entrer(Piece, apresMidi)$$

et

$$0.8 : NettoyerSol(Piece, Objet) \overset{Presence(Piece)}{\longleftarrow} Entrer(Piece, Temps)$$

Si on se trouve dans l'état instancié $Entrer(cuisine, apresMidi)$, les deux règles sont applicables, il va donc être nécessaire de déterminer laquelle suivre. En général on donne la préférence à la règle dont l'état initial est le plus spécifique. Dans cet exemple, le plus spécifique est le premier parce que le prédicat $Entrer(Piece, apresMidi)$ est subsumé par $Entrer(Piece, Temps)$.

Des extensions ont été réalisées à la version LoHMM telle que présentée par Kersting et coll. (2006). Par exemple, Natarajan et coll. (2008) on présenté un LoHMM hiérarchique afin de modéliser les activités pour la reconnaissance des activités. Cependant, ils n'ont pas expérimenté leur approche et l'efficacité de cette méthode dans des cas réels reste donc à

démontrer.

De manière générale, le plus grand avantage des LoHMM par rapport aux HMM est la généralisation des observations du domaine qui permet d'obtenir parfois une représentation plus compacte d'un problème complexe. L'inclusion des prédicats logiques permet une représentation plus riche des informations d'entrée et des états ainsi que la possibilité d'exprimer des relations. Par exemple, une même activité peut être réalisée dans des lieux spécifiques en fonction de la période de la journée et les actions qui la composent peuvent partager des objets. Cependant, comme pour les HMM, les LoHMM nécessitent une taille importante de données d'apprentissage pour estimer les probabilités. Étant un modèle dynamique, un LoHMM doit apprendre des paramètres pour des séquences d'événements ; par conséquent, l'exactitude de la prédiction de ce modèle dépend d'un corpus contentant plusieurs instances de toutes les séquences possibles qui peuvent représenter une information d'intérêt. Ceci limite donc son application pratique.

4.2.3.2 Réseaux logiques de Markov

Un réseau logique de Markov, *Markov Logic Network (MLN)* (Richardson et Domingos, 2006) est un modèle statistique qui intègre la logique du premier ordre et les réseaux de Markov.

Considérons F un ensemble de formules logiques du premier ordre et C un ensemble de constantes qui représentent les objets d'un domaine. Succinctement, une réseau logique de Markov L est constitué d'un ensemble de paires (f_i, w_i), où $f_i \in F$ et $w_i \in \mathbb{R}$ est un poids pour la formule f_i.

Lors du processus d'inférence qui sera détaillé dans la suite de cette section, tous les prédicats du réseau logique de Markov L sont instanciés et chacun devient une variable aléatoire appartenant à l'ensemble de variables X. Ces variables s'organisent dans un réseau de Markov, $M_{L,C}$, dont la structure est donnée par la règle suivante : il existe un arc entre deux sommets, $x_1, x_2 \in X$, si leurs prédicats d'origine se trouvent dans la même formule dans F. La figure montre le processus conduisant à l'obtention d'un réseau de Markov. L'objectif de l'inférence est de définir la probabilité $P(X = x)$, où x représente une configuration spécifique des variables X.

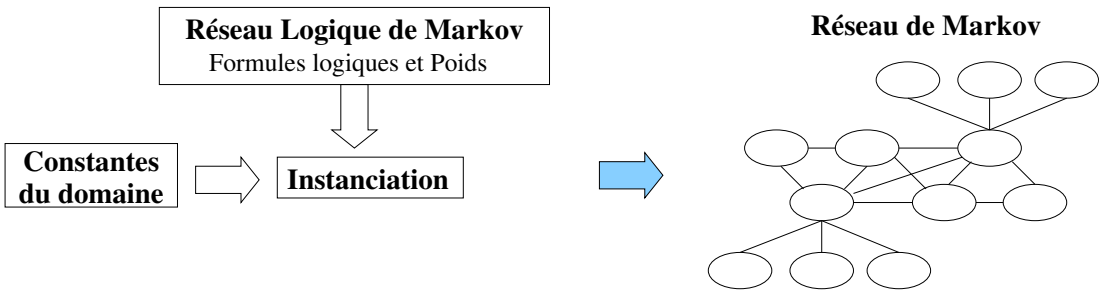


FIGURE 4.8: Obtention d'un réseau de Markov à partir d'un MLN

L'idée de ce modèle est que la probabilité d'une configuration spécifique des variables X

augmente si le nombre de formules logiques non satisfaites diminue. D'ailleurs, une configuration qui viole des formules logiques reste moins probable mais non impossible comme dans la logique de premier ordre.

Un exemple de la représentation d'un MLN, basé sur l'exemple donné de la section 4.2.2.5, peut être le suivant :

$1,23\ DureeCourte(int1) \wedge Utilisation(o,int1) \wedge SitueDans(o,Placard) \wedge ProduceSon(int1,Vaisselle)$
$\rightarrow SortirVaisselle(int1)$
$0,34\ DureeCourte(int1) \wedge DureeLongue(int2) \wedge Inclue(int2,int1) \wedge SortirVaisselle(int1) \wedge Localisation(Cuisine)$
$\rightarrow PreparerRepas(int2)$

Dans cet exemple, les variables commencent avec une lettre minuscule. Étant une représentation logique, elle est assez expressive comme celle de la logique de description. Il est spécifié que l'utilisation des objets du placard combinée à la présence d'un son de vaisselle sont une évidence de l'action de « sortir la vaisselle », et qu'une activité réalisée dans la cuisine qui inclut « sortir la vaisselle » est probablement la préparation d'un repas. Les poids indiquent la confiance dans ces règles. Il faut indiquer que pour garder la simplicité, nous avons enlevé les quantificateurs dans l'exemple. Pourtant, avec MLN il est possible de modéliser des quantificateurs existentiels et universels (Skarlatidis et coll., 2011). La figure 4.9 montre le réseau de Markov résultant si on considère que dans le domaine il existe seulement une constante pour chacune des variables.

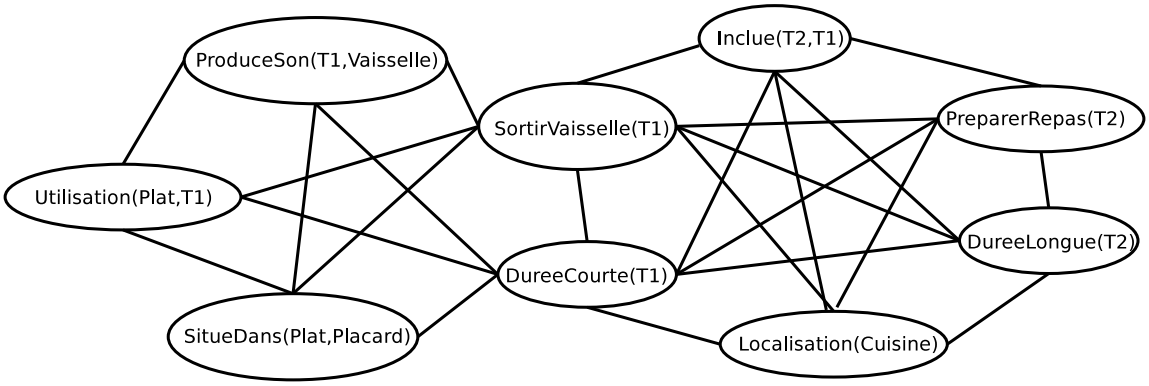


FIGURE 4.9: Réseau de Markov créé lors de l'inférence d'un MLN

Inférence dans le cas des Réseaux logiques de Markov

L'inférence commence par l'obtention des formules instanciées à partir du MLN. L'ensemble de variables du réseau de Markov, $M_{L,C}$, est constitué de tous les prédicats des formules instanciées. Pour construire la structure de $M_{L,C}$, il faut suivre le critère introduit au début de cette section : il existe un lien entre deux nœuds (prédicats instanciés si ils apparaissent ensembles au moins dans une formule instanciée. Ainsi, les atomes d'une formule instanciée forment un clique dans $M_{L,C}$.

Alors, la structure et la taille du $M_{L,C}$ dépendent non seulement des formules dans F, mais aussi du nombre de constantes dans C. Pour un ensemble différent de constantes, $M_{L,C}$ sera aussi différent. Dans l'exemple de la figure 4.9, si nous ajoutions une constante $Livre$, le $M_{L,C}$ serait plus complexe car il faudrait ajouter dans le graphe les variables $Utilisation(Livre,T1)$ et $SitueDans(Livre,Placard)$. Dans ce cas, il existerait les deux cliques suivantes, à partir

de la première formule, avec le même poids (1,23) :

- Clique 1 : $Utilisation(Livre, T1), SitueDans(Livre, Placard), ProduceSon(T1, Vaisselle), DureeCourte(T1), SortirVaisselle(T1)$
- Clique 2 : $Utilisation(Plat, T1), SitueDans(Plat, Placard), ProduceSon(T1, Vaisselle), DureeCourte(T1), SortirVaisselle(T1)$

La probabilité d'une configuration spécifique, $X = x$ (une assignation de vérité a toutes les variables de $M_{L,C}$) est donnée par la formule :

$$P(X = x) = \frac{1}{Z} exp\left(\sum_{f_i \in F} w_i n_i(x)\right) \tag{4.14}$$

où $Z = \sum_{x \in \chi} exp\,(P(X = x))$ est un facteur de normalisation, χ l'ensemble des états possibles. Z est donc simplement la somme des probabilités de chaque configuration possible. $n(x)$ est le nombre de formules instanciées qui sont vraies avec la configuration $X = x$.

Il faut noter que, si nous considérons x_k le sous ensemble de x contenant les prédicats de la clique k, la clique k produite par la formule i, nous pouvons définir la fonction de caractéristique :

$$\phi_k(x_k) = \begin{cases} e^{w_i}, & \text{si la valeur de vérité de } x_k \text{ satisfait sa formule d'origine, } f_i \\ 1, & \text{sinon} \end{cases} \tag{4.15}$$

Il est possible d'exprimer l'équation 4.14 comme suit :

$$P(X = x) = \frac{1}{Z} \prod_k \phi_k(x_k) \tag{4.16}$$

La formule 4.16 est équivalente à la formule 4.8 que nous avons définie pour les réseaux de Markov.

L'inférence peut être de type génératif ou discriminatif. Dans le premier cas, on cherche à calculer la probabilité d'une certaine configuration, $P(X = x)$. L'inférence discriminative estime la probabilité d'un sous-ensemble de variables (nommées variables cibles), $X_1 \in X$, étant donné la configuration, x_2, des autres variables comme évidence, $X_2 = x_2$, $X_2 \in X$. Donc, l'inférence discriminative est $P(X_1 \mid X_2 = x_2)$ et son résultat donne la probabilité de chaque instanciation possible de X_1. Il faut remarquer que les variables x_i du réseau de Markov sont toujours des variables binaires car elles représentent des prédicats instanciés dont les valeurs possibles sont vrai ou faux.

Ainsi, dans l'exemple précèdent d'inférence discriminative, si nous avons comme évidence un son de vaisselle dans l'intervalle de temps t_1 et que l'habitant se trouve dans la cuisine dans l'intervalle t_2, alors, si $t_1 \in t2$, il est possible d'inférer la probabilité de l'activité *Préparer Repas* dans l'intervalle t_2 :

$P(PreparerRepas(T2) \mid Inclue(T2, T1), ProduceSon(T1, Vaisselle), Localisation(Cuisine))$.

Ce calcul est fait par la sommation de la probabilité de chaque configuration x où le prédicat $PreparerRepas(t_2)$ est vrai. Pourtant, dans ce cas, l'ensemble des états possibles,

χ, est restreint aux configurations où les prédicats $Inclut(t_2, t_1)$,$ProduireSon(t_1, Vaisselle)$, $Localisation(Cuisine)$ sont aussi vrai. Z est donc calculé avec tous les instanciations possibles des prédicats restant : $SitueDans$,$Utilisation$, et $SortirVaisselle$.

Le calcul du facteur de normalisation Z rend l'inférence exacte dans un MLN complexe intraitable. Des méthodes d'approximation ont donc été proposées, par exemple, à travers l'approximation de type MCMC (*Markov chain Monte Carlo*). Actuellement, l'algorithme d'inférence pour MLN le plus performant est *MC-SAT* (Poon et Domingos, 2006) qui est une méthode MCMC utilisant un algorithme de satisfaisabilité booléen en sous-programme.

Apprentissage des Réseaux logiques de Markov

Les méthodes pour apprendre automatiquement un MLN sont généralement constituées de deux parties : l'estimation des poids des règles logiques et l'apprentissage de la structure du réseau (c'est-à-dire, les règles logiques).

Apprentissage des poids

D'une manière générale, l'apprentissage du poids des règles est indépendant de l'apprentissage de la structure. Il s'agit d'un processus d'optimisation d'une fonction de vraisemblance. Les méthodes d'estimation des poids peuvent être de type génératif ou discriminatif.

Dans l'approche générative, les poids peuvent être appris en maximisant la vraisemblance par rapport au corpus d'apprentissage x. Si la $i-eme$ formule a $n_i(x)$ instances qui la satisfassent avec les données en x, alors en utilisant l'équation 4.14, la dérivée de la log-vraisemblance par rapport au poids w_i est donné par l'équation 4.17.

$$\frac{\partial}{\partial w_i} \log P_w(X = x) = n_i(x) - \sum_{x'} P_w(X = x') n_i(x') \tag{4.17}$$

Un x' représente une configuration possible du corpus d'apprentissage x, donc la somme est effectuée sur l'ensemble des données possibles x' et $P_w(X = x')$ est $P(X = x')$ calculé en utilisant le vecteur $w = (w_1, \ldots, w_i, \ldots)$. En d'autres termes, le i^{eme} composant du gradient est donné par la différence entre le nombre d'instances satisfaisant la i^{eme} formule (c.-à-d., $n_i(x)$) et son espérance ($\sum_{x'} P_w(X = x') n_i(x')$). La vraisemblance est une fonction convexe. La maximisation de cette vraisemblance est donc faite par un processus itératif qui converge vers un optimal de w lorsque le gradient s'approche de zéro. Malheureusement, sauf dans les cas très simples, le calcul de la deuxième partie de la soustraction dans l'équation 4.17 n'est pas solvable.

Pour estimer les poids, Richardson et Domingos (2006) ont alors proposé une méthode qui optimise la *pseudo*-vraisemblance (Besag, 1975), $P^*_w(X = x)$, donnée par l'équation :

$$P^*_w(X = x) = \prod_{l=1}^{n} P_w(X_l = x_l \mid MB_x(X_l)) \tag{4.18}$$

où $MB_x(X_l)$ est la couverture de Markov de la variable X_l (l'ensemble de nœuds voisins dans la représentation graphique).

Même si cette méthode est plus rapide, elle peut être inefficace dans certaines circonstances car les solutions sont sous-optimales.

Dans le cadre de nos recherches, nous pouvons toujours identifier quels sont les prédicats dont la probabilité nous intéresse, et celles qui nous servent d'évidence. Par conséquence, nous allons travailler avec l'apprentissage discriminatif. En ce cas, les prédicats instanciés sont divisés en deux parties : X, les évidences ; et Y, les cibles. Comme dans le cas précédent, il est ici également nécessaire d'optimiser le gradient de la log-vraisemblance qui est donné par :

$$\frac{\partial}{\partial w_i} \log P_w(y \mid x) \;=\; n_i(x,y) - \sum_{y'} P_w(y' \mid x)\, n_i(x,y') \;=\; n_i(x,y) - E_w[n_i(x,y)] \tag{4.19}$$

Dans ce cas aussi l'espérance, E, n'est pas solvable parce que cela demande le calcul de la probabilité pour chaque configuration y' de l'ensemble de variables cibles. Pour cela, il est possible d'utiliser l'inférence MAP (*maximum a posteriori*) qui consiste à trouver l'état le plus probable, y^*, d'un ensemble de variables cibles(Y) sachant l'état des variables d'évidence (X). Le MAP est estimé par des méthodes MCMC, par exemple *Gibb sampling* (Casella et George, 1992). Il est possible d'estimer l'espérance par le nombre $n_i(x, y^*_w)$ dans l'état du MAP ayant $y^*_w(x)$. Ainsi, l'estimation du gradient demande seulement l'inférence du MAP. Singla et Domingos (2005) ont proposé l'utilisation de l'algorithme itérative *voted perceptron* dans lequel, dans chaque itération, l'estimation MAP est faite par l'algorithme *MaxWalkSat*. Dans l'état de l'art, la méthode la plus utilisée est *Scaled Conjugate Gradient* qui est aussi basée sur le même principe de *voted percepton* pour trouver les poids optimaux de façon itérative (Lowd et Domingos, 2007).

Apprentissage de la structure

Les algorithmes d'apprentissage de structure à partir des données peuvent être classifiés en 2 approches : descendantes et ascendantes. Dans le premier cas, l'apprentissage est initialisé avec un ensemble de règles vide ou un ensemble défini *a priori* et les prédicats sont évalués et ajoutés à cet ensemble à chaque itération. Cependant, si les règles sont très complexes, l'espace des raffinements possibles peut devenir insolvable. Dans l'approche ascendante (Kok et Domingos, 2005), le processus cherche des hypothèses, chaque fois plus générales, qui expliquent les données d'apprentissage. Dans ce cadre, Kok et Domingos (2010) ont proposé un algorithme d'apprentissage automatique de structure qui génère des règles candidates et mesure leur pertinence par une estimation du maximum de log-vraisemblance. connaissant les données d'entrées. Cependant, dans plusieurs domaines, comme la reconnaissance d'activités, les prédicats devant être inférés sont désignés préalablement. Par conséquent, un modèle discriminant est nécessaire. Dans ce type d'apprentissage, l'objectif est de trouver un modèle pour estimer la probabilité conditionnelle d'un sous-ensemble de variables étant donnée un autre sous-ensemble.

Des autres méthodes pour l'apprentissage de structure consistent à utiliser les graphes de prédicats (Dinh et coll., 2011) où la programmation logique inductive (Huynh et Mooney, 2008). Dans le cas de l'ILP, les auteurs utilisent l'algorithme MDIE (*Mode-Directed Inverse Entailment*) (Muggleton, 1995) pour apprendre des clauses de Horn. Puis, ils emploient une

méthode pour estimer les poids de l'ensemble des règles obtenues.

Enfin, les règles peuvent également être définies manuellement à partir de connaissances expertes sur le domaine d'application. L'intérêt de l'ILP est qu'une connaissance d'arrière plan ou *background knowledge* peut être fournie en entrée de l'algorithme d'induction et que celui-ci peut faire appel à une connaissance *a priori*. Cette injection de connaissances expressives est beaucoup moins systématisée dans les méthodes d'induction des modèles purement probabilistes.

Comme pour le LoHMM, le MLN offre un cadre de travail très riche pour représenter des relations complexes entre les entités d'un domaine. Cependant, le MLN rend possible l'expression de n'importe quelle connaissance du domaine sous forme de clause de Horn alors que le LoHMMne permet de modéliser qu'un aspect spécifique tel que les transitions entre états. Étant un modèle très général, même les relations temporelles d'une séquence peuvent être représentées par des prédicats logiques. On peut donc modéliser un HMM par un MLN (Artikis et coll., pear). Par ailleurs, l'apprentissage de la structure du MLN peut être améliorée par l'inclusion de règles *a priori* issues de connaissances expertes du domaine d'application.

4.2.4 Discussion des méthodes d'inférence

Ce rapide état de l'art montre qu'il existe une littérature foisonnante sur les différents modèles de raisonnement pouvant être appliqués à l'intelligence ambiante. Une approche toujours populaire est basée sur la représentation sous forme de règles logiques que cela soit dans le cas de systèmes experts ou de raisonnements basés sur une description logique d'une ontologie. Cependant, les auteurs de ces approches reconnaissent que l'incertitude ou l'imprécision devraient être pris en compte au niveau de la base de connaissances ou des données. Inversement, les approches statistiques ont reçu beaucoup d'attention dans le cadre de la reconnaissance de situations car ils peuvent traiter l'incertitude inhérente aux données des maisons intelligentes. Par ailleurs, des algorithmes efficaces existent pour apprendre les modèles gérant l'incertitude. Par exemple, les approches à base de réseaux Bayésiens ont été considérablement utilisées de même que les SVM. Toutefois, ces approches ne réussissent pas à représenter et tirer parti efficacement des relations entre les informations. Les approches statistique-relationnelles abordent ces deux problèmes et semblent très pertinentes dans les cas où il existe des relations complexes entre les informations qui peuvent être utiles à amélioration des résultats de l'inférence.

De cet ensemble de méthodes d'inférence et de leurs applications dans les environnements perceptifs, il apparaît clairement que le choix du modèle d'inférence dépend du type de données à traiter. Par exemple, une approche probabiliste semble adaptée aux données brutes des capteurs tandis qu'une approche mêlant modèles probabilistes et représentations logiques semble plus adaptée à reconnaissance de situations. En conséquence, nous utilisons des modèles d'inférence différents en fonction des tâches d'inférences, des données d'entrées et des niveaux d'abstraction plutôt que d'utiliser une approche unique non

nécessairement optimale dans chaque cas. Bien que cela induise des modèles différents, les informations manipulées par les modèles sont unifiées à travers une base de connaissances commune qui permet de représenter les informations des sources d'entrée jusqu'aux décisions finales du contrôleur intelligent.

4.3 Localisation de l'habitant

La localisation d'une personne est souvent l'élément primordial du contexte dans les systèmes sensibles au contexte (Schilit et coll., 1994; Brown et coll., 1997). C'est également le cas dans notre application pour la prise de décision. Par exemple, si une personne seule dans un habitat prononce la commande vocal « allume la lumière », il est nécessaire de localiser la personne pour déduire la lampe à activer.

Les approches présentées dans la littérature reposent souvent sur l'utilisation de la vidéo (Friedland et coll., 2010). Cependant, pour limiter les coûts et pour respecter la vie privée, seuls les capteurs domotiques sont considérés dans notre approche. Ces sources (ici capteurs de présence, capteurs de contact, et microphones) sont ambiguës (par exemple, l'observation d'un même événement par plusieurs capteurs éloignés), ont une dimension temporelle (par exemple, dans le couloir à `15:23:45`) et dépendent de l'espace physique (p.ex., nombre de pièces, réverbération). Il devient donc essentiel de mettre en place un système de localisation multisource qui tirerait partie des redondances et des complémentarités des sources choisies.

Nous avons donc mis en œuvre une méthode de fusion de données en utilisant des sources d'informations non visuelles (c.-à-d. sans caméra), événementielles et indirectes (c.-à-d. sans capteur porté par la personne). L'approche que nous avons adoptée pour localiser une personne se base sur un réseau dynamique (cf. section 4.2.1.2). Il convient de noter que l'approche considère un seule personne dans l'appartement ni plus (c.-à-d., un groupe), ni moins (c.-à-d., le cas appartement vide). Dans ce réseau les événements des capteurs activent des hypothèses de localisation qui se périment au cours du temps. Ces hypothèses prennent en compte les activations précédentes, l'incertitude des événements et la durée de validité d'un événement. Dans notre approche, le réseau est dynamique dans le sens où il évolue en fonction des données, et où il prend en compte des connaissances *a priori* sur l'environnement dans lequel évolue l'habitant. Une contribution importante de notre approche est la prise en compte de l'aspect temporel par la diminution de l'activation des nœuds au cours du temps, qui modélise la diminution de la certitude sur la localisation de l'habitant dans le temps.

4.3.1 Formalisation

La méthode que nous avons conçue utilise les définitions suivantes :

Définition 5 (Événement) *Occurrence d'un phénomène physique dans l'environnement qui peut être détecté par un capteur.*

Les événements sont par exemple, un claquement de porte, un déplacement, un appel téléphonique, une parole…

Définition 6 (observation) *Une observation o_{t_n} est une donnée structurée générée par un capteur ayant réagi à un événement e_{t_n} au temps t_n avec $n \in N$. À chaque observation o est associé le type de capteur $o.type$, l'origine du capteur $o.capteur$, et le rapport signal sur bruit (RSB) $o.rsb$ dans le cas de signaux numériques.*

Définition 7 (observations simultanées) *Deux observations a_{t_n} et b_{t_k} sont dites simultanées si $t_n = t_k$ et $a \neq b$.*

Définition 8 (hypothèse de localisation) *Étant donné un ensemble de localisations L défini par $L = \{Loc_1, \ldots, Loc_R\}$, et i, $i \in N$ tel que $h^i_{t_n} = Loc_i$, alors $h^i_{t_n}$ est le nœud hypothèse que l'habitant se trouve à la i^e localisation à l'instant t_n. Les hypothèses sont générées uniquement à partir des observations au temps t_n.*

Définition 9 (contexte d'occupation) *Étant donné un ensemble d'occupations P défini par $P = \{Piece_1, \ldots, Piece_R\}$, et i, $i \in N$ tel que $c^i = Piece_i$, alors c^i est le nœud de contexte d'occupation de la i^e pièce. La valeur d'activation du contexte c^i varie en fonction du temps et des hypothèses.*

Définition 10 (poids de relation) $w \in [0,1]$ *est la* force *de la relation entre deux nœuds du réseau. Ainsi, w_{o,h^i} est le poids du lien entre une observation o et l'hypothèse que l'habitant se trouve à la i^e localisation. w_{h^i,c^j} est le poids du lien entre l'hypothèse que l'habitant est à la i^e localisation et le contexte j.*

Définition 11 (fonction d'oubli) *La fonction d'oubli $f(t_n, t_{n-1}) = e^{-\frac{\Delta_t}{\tau}}$, avec $\Delta_t = t_n - t_{n-1}$, représente la décroissance d'un contexte au cours du temps. Elle permet de conserver une mémoire à court terme des contextes. Plus les observations seront espacées au cours du temps (p.ex., $\Delta_t > 3 \times \tau$), plus rapidement les contextes seront oubliés.*

4.3.2 Architecture du réseau dynamique

Le réseau dynamique que nous avons conçu est organisé en deux niveaux : le premier niveau correspond aux *hypothèses de localisation* générées à partir de l'observation d'un *événement*, et le deuxième niveau représente les *contextes d'occupation* de chaque pièce ; le poids d'activation des différents contextes indique la localisation la plus probable connaissant les événements précédents.

4.3.2.1 Évolution du réseau dynamique

L'algorithme fonctionne ainsi :

1. pour toute nouvelle observation $o^k_{t_n}$, un nouveau nœud est créé ;

2. des nœuds hypothèses $h^i_{t_n}$ sont alors créés pour tout $i \in [1;r], r \leq R$ avant d'être connectés aux observations $o^k_{t_n}$ avec les poids w_{o^k,h^i} ;
3. les hypothèses $h^i_{t_n}$ sont connectées aux contextes de localisation c^j avec les poids w_{h^i,c^j} ;
4. les activations sont propagées des $o^k_{t_n}$ aux $h^i_{t_n}$ et l'activation des $h^i_{t_n}$ est calculée ;
5. les activations sont propagées de $h^i_{t_n}$ à c^j et l'activation de c^j est recalculée ;
6. le c^j avec la plus forte activation devient la localisation courante ;
7. tous les nœud $h^i_{t_n}$ et $o^k_{t_n}$ sont supprimés du réseau.

Un exemple de réseau dynamique est donné par la figure 4.10. À l'instant t_{n-2}, l'événement $e_{t_{n-2}}$ est capté par un capteur qui génère une observation $o_{t_{n-2}}$ dont 3 hypothèses sont déduites : $h^1_{t_{n-2}}$ avec une relation de 0,1 vers le contexte c^1, $h^2_{t_{n-2}}$, avec 0,6 vers c^2 et $h^3_{t_{n-2}}$ avec 0,3 vers c^3. S'il n'y a pas eu d'événement antérieur, c^2 est la localisation la plus probable. À l'instant t_{n-1}, l'événement $e_{t_{n-1}}$ est observé par deux capteurs. Les nœuds générés à t_{n-2} disparaissent, seul les contextes sont conservés. Si ces contextes sont encore actifs, leurs activations seront pondérées par $f(t_{n-1}, t_{n-2})$ auxquels on ajoute les activations générées par les nouvelles hypothèses. Dans le cas de c^2, l'activation sera incrémentée de $0,1+0,1=0,2$ car l'hypothèse $h^1_{t_{n-1}}$ est due à 2 événements simultanés $o^1_{t_{n-1}}$ et $o^2_{t_{n-1}}$. Dans le cas de c^4, l'hypothèse $h^2_{t_{n-1}}$ est due à une observation unique ; l'incrément sera égal au poids soit 0,8. La méthode s'appliquera ensuite de la même manière à l'instant t_n, etc.

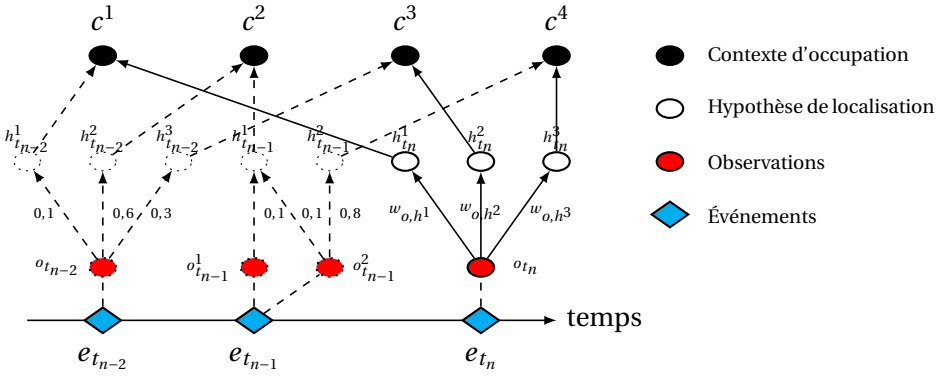


FIGURE 4.10: Exemple de réseau dynamique

4.3.2.2 Propagation de l'activation

L'activation recueillie par un nœud est classiquement définie (Crestani, 1997) par la formule

$$n_i(t) = \sum_{i \neq j} w_{i,j} \times A^j(t) \tag{4.20}$$

où $w_{i,j}$ est le poids, j correspond à un voisin de i et $A^j(t)$ est l'activation de ce voisin au temps t. Un nœud qui a été activé par un voisin ne peut communiquer de nouveau cette activation à ce voisin. Dans notre cas, les activations sont toujours initiées par une observation o qui a à chaque fois une activation maximale $A^o(t) = 1$ quel que soit son type et se propage de bas en haut dans le réseau dynamique. Une fois l'activation $n(t)$ provenant des voisins

accumulée, l'activation de sortie du nœud doit être calculée. Elle diffère selon le niveau des nœuds. Pour les hypothèses de localisation, l'activation $A^{h^i}(t) \in [0,1]$ est calculée selon la formule 4.21 :

$$A^{h^i}(t_n) = n_i(t_n) = \sum_{o \in O_{t_n}} w_{o,h^i} A^o(t_n) \tag{4.21}$$

où O_{t_n} est l'ensemble des observations simultanées à t_n liées à h^i, et $\sum_{o \in O_{t_n}} w_{o,h^i} \leq 1$. Pour les contextes d'occupation de pièce, l'activation de sortie résulte de l'accumulation des activations provenant des hypothèses et de l'activation précédente, pondérée par un facteur d'oubli. L'équation 4.22 décrit la loi d'activation du contexte de localisation A^{c^i} suite à une activation externe au temps t_n.

$$A^{c^i}(t_n) = n_i(t_n) \times [M - e^{-\frac{\Delta t}{\tau}} A^{c^i}(t_n - \Delta t)] + e^{-\frac{\Delta t}{\tau}} A^{c^i}(t_n - \Delta t) \tag{4.22}$$

où $A^{c^i}(t_n - \Delta t)$ est l'activation précédente, $M = 1$ est l'activation maximale, et $e^{-\frac{\Delta t}{\tau}}$ est la fonction d'oubli. Ainsi, si aucun événement n'apparaît pendant $5 \cdot \tau$ secondes, l'activation des contextes pourra être considérée comme nulle. L'introduction de M permet de maintenir toutes les activations entre 0 et 1.

4.3.2.3 Calcul des relations entre les différentes couches du réseau

Le réseau dynamique étant constitué de deux niveaux bien définis (cf. figure 4.10), deux types de relations existent, la relation *observation-hypothèse* et la relation *hypothèse-contexte*. Les liens entres les différentes couches dépendent fortement de l'application et de l'environnement considéré.

Relation hypothèse-contexte La relation *hypothèse-contexte* est dans notre cas une relation univoque, car une hypothèse de localisation n'est liée qu'à une seule pièce. Il s'agit d'un choix expérimental car certaines hypothèses sur des pièces faiblement séparées (p.ex. salon/chambre) pourraient activer plusieurs contextes de pièce. Par ailleurs, l'application visée étant la localisation d'un habitant dans une pièce, tous les nœuds contextes de pièces sont créés à l'initialisation du réseau et en font partie en permanence.

Relation observation-hypothèse La relation *observation-hypothèse* est unidirectionnelle et de type 1-n (un vers plusieurs). Le poids et les hypothèses générés varient en fonction de la source des observations. Cette relation est représentée en forme de probabilité en considérant deux types d'information : des informations dynamiques contenues dans une observation O_n, et celles qui sont statiques car elles proviennent des connaissances *a priori* $\mathscr{C}$ sur les capteurs. Le poids de cette relation est donnée par l'équation 4.23.

$$w_{o,h^i}(t_n) = P(loc = i \mid O_n, \mathscr{C}) \tag{4.23}$$

Les résultats de l'évaluation de cette méthode sont présentés dans la section 6.4.

4.4 Reconnaissance d'activités

Un autre élément important du contexte pour la prise de décision est l'activité de la personne. Par exemple, si la personne prononce la commande vocale « allume la lumière », l'action sera différente si elle se réveille en pleine nuit (dans ce cas, la meilleure action serait d'allumer la lampe de chevet avec une faible intensité) ou si elle fait le ménage (dans ce cas, la meilleure action serait d'allumer le plafonnier avec une forte intensité). Une tâche importante du contrôleur intelligent est donc d'inférer l'activité de la personne à chaque instant.

La reconnaissance automatique de l'activité humaine (manger, discuter, regarder la télévision, etc.) peut être définie comme l'identification d'un ensemble d'actions atomiques temporellement liées entre elles (prendre un ustensile de cuisine, s'allonger, etc.). Ceci implique d'abstraire les signaux bruts en symboles (propositions) étiquetés temporellement (p.ex. : claquement de porte à 11h32), et de reconnaître parmi ceux-ci des signatures d'activités particulières. Par exemple, les mouvements détectés dans la chambre à coucher peuvent faire partie de l'activité « se lever » qui elle même peut faire partie d'un plan de journée (p.ex. : matinée du dimanche). La reconnaissance automatique d'activité est l'un des secteurs de recherche les plus actifs et les plus ambitieux en raison de la quantité importante de bruit dans les données et de la difficulté de modélisation des situations ; pour une même personne, une activité peut se dérouler de nombreuses façons différentes.

Il existe plusieurs niveaux de granularité de définition d'activités humaines, dans notre cas nous nous basons sur un sous ensemble des activités de la vie quotidienne catégorisées par Katz (1983). Ceci nous conduit à définir l'activité comme suit :

Définition 12 (Activité) *Une activité est une tâche de la vie quotidienne, telle que dormir, s'habiller, se nourrir, communiquer, ... que la personne effectue sur un intervalle de temps donné.*

Dans cette section, nous présentons la méthode à base de MLN (cf. section 4.2.3.2) que nous avons employée pour effectuer la reconnaissance d'activité de la vie quotidienne dans un environnement perceptif. Nous présentons également les résultats préliminaires obtenus.

4.4.1 Reconnaissance d'activités par MLN

L'approche adoptée part du principe qu'une activité est un ensemble d'événements successifs qui génèrent des observations sur des capteurs placés dans l'environnement. Nous faisons l'hypothèse que ces observations sont des traces des activités et que celles-ci peuvent être décrites par des statistiques de variables prédéfinies estimées sur des fenêtres temporelles plus courtes que la durée minimale d'une activité. Ce « fenêtrage » ne permet pas d'obtenir un modèle très précis des activités mais les activités détectées de cette manière seront suffisamment informatives pour donner du contexte à la prise de décision.

La méthode employée pour reconnaître les activités au fur et à mesure de l'arrivée des données (c.-à-d. en-ligne) suit le schéma de la Figure 4.11. Les données brutes extraites des capteurs sont composées de valeurs symboliques instantanées (p.ex. : détecteur infra-rouge)

ou persistantes (p.ex. : interrupteur), de série temporelles (p.ex. : température) et de signaux continus (p.ex. : microphones). Un prétraitement permet d'extraire des informations de haut niveau comme la parole, les sons, la localisation et le niveau d'activité de l'habitant. Toutes les données sont abstraites en vecteurs de description, chaque vecteur résumant une fenêtre temporelle. Le traitement de ces vecteurs par les classificateurs permet de décider du type d'activité en cours le plus probable. Les modèles de classification sont appris de manière supervisée.

4.4.1.1 Information inférée

Les données brutes provenant de l'habitat intelligent contiennent de l'information qu'il est nécessaire d'extraire en vue de la reconnaissance d'activité. Deux types de traitements sont considérés ici : la localisation de la personne — qui est une information primordiale pour la reconnaissance d'activité —, et la détection de parole — qui est importante pour reconnaître les activités de communication.

(a) **Extraction de parole :** Dans notre approche, les événements audio sont détectés, en temps réel, par un algorithme à seuil adaptatif basé sur une transformée en ondelettes (Vacher et coll., 2010a). Les événements sont ensuite classés en parole ou son de la vie courante (p.ex. : chute d'objet) par un classificateur utilisant un modèle de mélange de gaussiennes (24 gaussiennes) appris par l'algorithme *Expectation Maximisation* (EM). Les microphones étant omnidirectionnels, un son peut être enregistré en même temps par plusieurs microphones placés dans plusieurs pièces ; ces observations simultanées sont repérées. Pour chaque observation sonore, le rapport signal sur bruit (RSB) est calculé.

(b) **Localisation :** La méthode que nous avons mise en place est décrite dans la section 4.3.

(c) **Niveau d'activité (Agitation) :** Dans une maison intelligent il s'agit d'une mesure de la fréquence des actions de l'habitant. Cela permet de discriminer parmi les activités qui comportent plusieurs actions (comme le rangement), de celles qui sont plus calmes (ex. dormir). Nous estimons le niveau d'activité à partir des données des capteurs infrarouge, contacts de porte, et sons. Ces données sont fusionnées avec un modèle linéaire. Le niveau d'activité — une valeur numérique — ne doit pas être confondu avec l'activité— qui est une catégorie des activités de vie quotidienne.

4.4.1.2 Calcul des vecteurs d'attributs

Les traces provoquées par une activité humaine et générées par les capteurs sont difficilement généralisables étant donné la très grande variété de réalisations d'une même tâche inter- et intra-personne. Nous considérons donc des fenêtres temporelles à l'intérieur desquelles les données brutes et inférées sont résumées. Au total, 31 attributs ont été extraits pour chaque fenêtre temporelle. Ils sont résumés dans le tableau 4.2. La classe rejet est composée de fenêtres dans lesquelles aucune des activités prédéfinies n'a été exécutée.

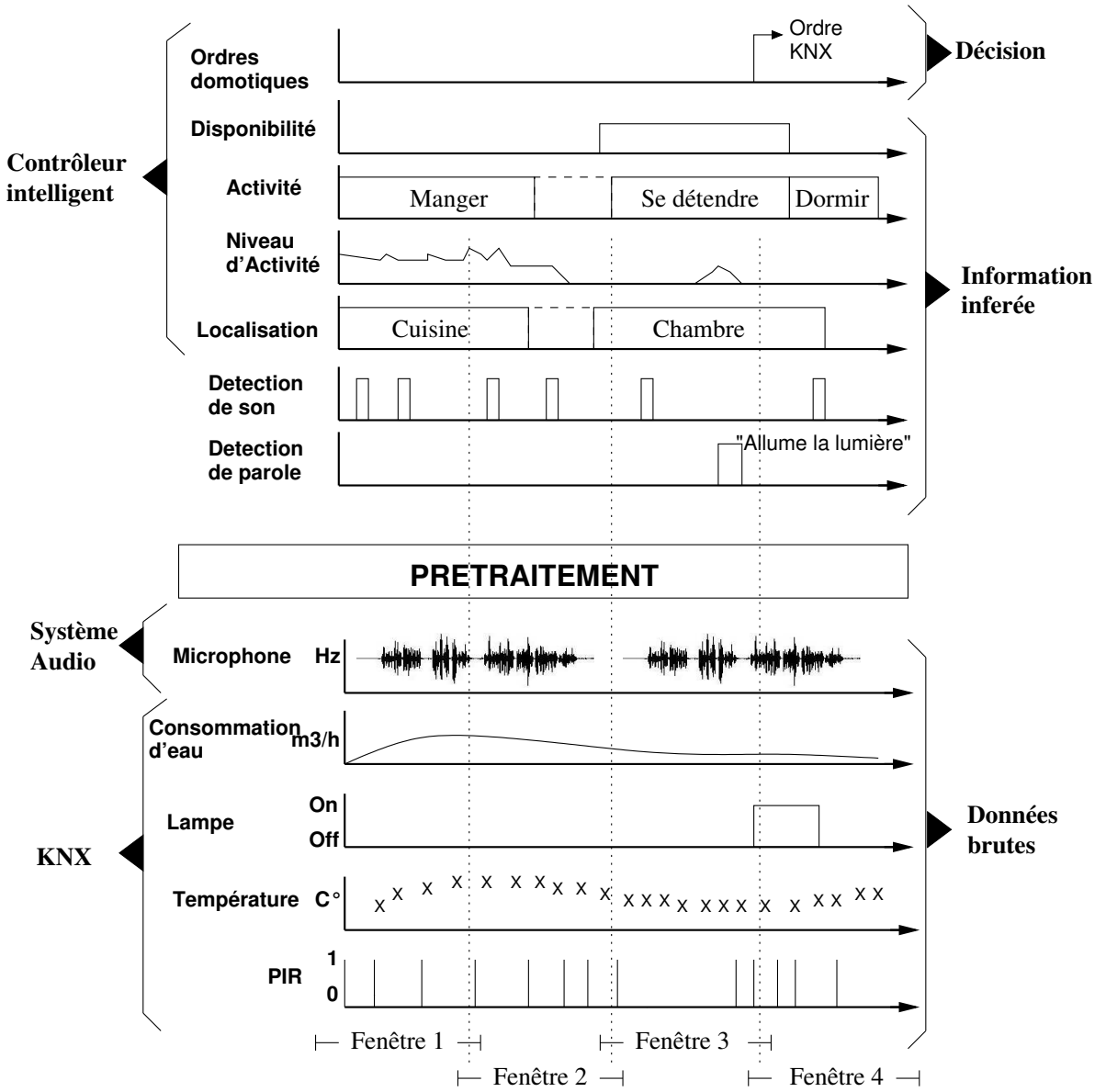


FIGURE 4.11: Fenêtrage temporel utilisé pour l'obtention des attributs à partir des données fournies par les capteurs.

4.4.2 Adaptation des réseaux logiques de Markov à la reconnaissance d'activités

Nous avons testé trois approches d'apprentissage de la structure d'un MLN pour la reconnaissance d'activités :

1. Le **MLN naïf** utilise une liste de règles qui modélisent la capacité de chaque valeur possible d'un attribut pour classifier une activité. Cette capacité est modélisée indépendamment des autres valeurs et attributs. Formellement, ces règles ont la structure suivante
   $attribut_i(X, V_i) \rightarrow classe(X, V_c)$
   où les variables X, V_i, et V_c représentent respectivement : la fenêtre temporelle à classifier, une valeur du i^e attribut, et une valeur de la classe cible (l'activité). Ainsi, avec n attributs, si $|V_i|$ est la cardinalité de l'ensemble des valeurs possibles de l'attribut i, le

Attribut	Nombre	Information
PourcentageLoc_x	4	Pourcentage de temps dans la fenêtre temporelle ayant x comme localisation
PredominantLoc	1	Localisation avec le plus grand pourcentage
DerniereLoc	1	Dernière localisation avant le début de la fenêtre temporelle
TempsDansLoc	1	Temps en seconds dans la localisation prédominant
ActDesactFenetre_y	6	Nombre d'activations/désactivations du contact de la fenêtre y
ActDesactPorte_w	4	Nombre d'activations/désactivations du contact de la porte w
ActDesactLumiere_z	6	Nombre d'activations/désactivations de l'interrupteur de la lumière z
ActivationDeactivation CommDoor_f	5	Nombre d'activations/désactivations du contact du placard f
DetectionPIR_x	2	Nombre d'activations du capteur de mouvement dans la localisation x
AmbientSensor	13	Variation de la valeur entre les fenêtres de : CO2, température, Humidité, luminosité, débit d'eau et électricité.
Power_LastUse	3	Identifiant de la dernière utilisation d'une prise électrique ou d'un interrupteur.
Pourcentage_Sound	1	Pourcentage de temps occupé par le son dans la fenêtre temporelle
Pourcentage_Speech	1	Pourcentage de temps occupé par la parole dans la fenêtre temporelle
Son_m	7	Nombre d'événements de sons détectés par le microphone m
Parole_m	7	Nombre d'événements de parole détectés par le microphone m
PourcentageAgitation_	6	Nombre d'événements par fenêtre dans les categories : localisation, portes, électricité, eau, sons et parole
TotalAgitation	1	Somme de PercentageAgitation_ dans la fenêtre temporelle
Classe	1	Préparer et prendre un repas, rangement, hygiène, communication, s'habiller/se déshabiller, dormir, se détendre, classs rejet

TABLE 4.2: Attributs utilisés pour la classification

nombre de règles dans le modèle est donné par : $\sum_i^n |V_i| \cdot |V_c|$.

De plus, la relation temporelle entre les instances a été modélisée à travers la règle suivante :

$previous(X, V_c) \rightarrow classe(X, V_c)$

Cette règle est pertinente car normalement les activités de vie quotidienne suivent une certaine routine dans la journée d'un habitant. Nous avons nommé cette approche « naïve » car elle considère les attributs comme étant conditionnellement indépendants sachant la variable cible, comme dans les réseaux bayésiens naïfs.

2. Le **MLN+PLI1** où l'apprentissage de la structure s'effectue à travers la programmation logique inductive (voir 4.2.2.3). L'algorithme MDIE proposé par Muggleton (Muggleton, 1995), implémenté dans le logiciel PROGOL, génère une hypothèse sous forme de programme logique pour expliquer des prédicats cibles à partir des connaissances introduites sous forme de règles et une liste d'exemples positifs et négatifs. Dans ce cas les règles que nous avons apprises sont des clauses de Horn :

   $attribut_1(X, V_i) \wedge attribut_2(X, V_i) \wedge ... \wedge attribut_m(X, V_i) \rightarrow classe(X, V_c)$

   La valeur m n'est pas fixe, elle varie entre 1 et 4 pour chaque règle. Nous avons trouvé que les règles logiques ayant cette taille donnent les meilleurs résultats. Le principale avantage de ce modèle par rapport au précèdent est d'obtenir un nombre de règles

beaucoup plus réduit, et par conséquence d'avoir un modèle moins complexe. Par ailleurs, ILP permet d'obtenir des connaissances par rapport a la relation de plusieurs attributs pour classifier une activité. L'utilisation de la PLI pour apprendre la structure d'un MLN a été étudiée par Huynh et Mooney (2008) sur des données artificielles.

3. Le **MLN+PLI2** qui implique une variation dans l'algorithme d'apprentissage MDIE. Dans cette variation, chaque exemple positif est utilisé indépendamment des autres pour générer des règles. Ceci conduit à un plus grand nombre des règles apprises dont les pertinences peuvent être pondérées lors de la phase d'apprentissage des poids.

Une fois que les règles ont été définies, l'algorithme *Scaled Conjugate Gradient* (Lowd et Domingos, 2007) a été employé pour apprendre leur poids sur le corpus d'apprentissage.

Les résultats de l'évaluation de cette méthode sont présentés dans la section 6.5.

4.5 Reconnaissance de situations

Dans les systèmes informatiques, la reconnaissance de situations (Dousson et coll., 1993; Ye et coll., 2012) consiste à identifier un état des lieux qui informe sur la décision/réaction à entreprendre. La définition d'une situation est très subjective et extrêmement dépendante de l'application et des sources d'informations disponibles. Ye et coll. (2012) donnent une distinction entre la reconnaissance d'activité et la reconnaissance de situation. Pour eux, une situation contient une structure temporelle beaucoup plus riche qu'une activité. Par exemple, certaines situations ne peuvent se produire qu'à certains moments de la journée : une entrée dans une banque est suspicieuse si elle est effectuée en pleine nuit ou durant le week-end, mais beaucoup moins si elle se produit pendant les heures d'ouverture.

4.5.1 Les situations dans le cadre de l'habitat intelligent

Une situation dépend exclusivement de la connaissance de l'état courant du système à partir des données obtenues des sources d'information. Les informations contextuelles à utiliser pour la reconnaissance de situations sont la localisation, l'activité et l'heure. Chacune de ces informations est inférée avec des méthodes traitant l'incertitude. La reconnaissance de son et paroles, sont aussi des sources d'information qui peuvent être affectés par l'incertitude ; et en général, c'est le cas de la plupart des sources d'informations et de toutes les informations de haut niveau qui demandent une méthode d'inférence. À partir de ces informations ayant un certain degré d'incertitude, il est possible de développer des modèles de reconnaissance de situations, ayant pout base des approches logiques, qui utilisent uniquement les sources d'informations les plus fiables et laissent le traitement de l'incertitude à l'étape de prise de décision.

En effet, à la différence d'autres travaux (Ricquebourg et coll., 2007) dans lesquels le système de contrôle domotique est réactif et la décision d'exécuter une action est directement définie comme la conséquence d'une condition (dans ce cas, la situation reconnue) ; nous avons séparé ces deux étapes. La reconnaissance de situation est effectuée en amont de la

décision qui est la seule étape de traitement à utiliser le contexte. La reconnaissance de situations ne prend en compte que les informations jugées fiables, puis la décision s'adapte au niveau de certitude qui caractérise l'état du système courant selon les informations contextuelles. Ainsi, deux instances de la même situation peuvent conduire à des décisions différentes selon l'information contextuelle fournie dans chaque cas.

Dans notre application, nous avons prévu deux types de reconnaissance de situation :

- **Ordres Vocaux.** C'est le type de reconnaissance de situation le plus simple à traiter. Les phrases prononcées par l'habitant sont détectées par un système de reconnaissance vocale qui identifie les ordres domotiques et les envoie au contrôleur intelligent. Dans ce cas, la reconnaissance de situations permet juste de traiter l'ordre selon le contexte.
- **Situations prédéfinies.** Il s'agit de situations qui sont établies comme des états du système qui représentent un risque pour le confort ou la sécurité de l'habitant. La définition de la situation décrit les valeurs des sources d'information impliquées où l'assignation d'une valeur correspond à un événement. Cette description peut également contenir des relations temporelles entre les événements. Par exemple, une situation peut être spécifiée pour détecter quand l'habitant laisse accidentellement la porte principale ouverte avant de passer plusieurs minutes dans sa chambre.

Toutes les situations sont associées à une liste de commandes qui doivent être exécutées pour conserver le bien-être de l'habitant. C'est le module de décision qui gère la façon d'exécuter les commandes.

4.5.2 Application de SWRL pour la reconnaissance de situations

Dans le cas des situations de perte de confort ou de danger, une situation peut être considérée comme un motif temporel de l'état du système qui est donné par la base de faits. Les ontologies offrent un moyen approprié pour la reconnaissance de situations puisqu'elles peuvent stocker l'ensemble des faits ayant lieu dans l'environnement, les interactions avec l'habitant, et les connaissances procédurales pour faire des inférences. De plus, la dimension temporelle peut être prise en compte à travers des rôles entre les événements, tels que *precedent* et *suivant*, qui peuvent être généralisés par des propriétés d'enchaînement pour obtenir des relations plus générales telle que *avant* et *apres*. Les situations peuvent ainsi être définie par règles logiques où les relations temporelles sont spécifiées par des prédicats ayant des événements en paramètres.

Nous avons choisi SWRL pour modéliser les situations. Même si cette méthode est limitée à l'utilisation des *DL-safe rules*, elle permet la définition de relations plus complexes que l'utilisation de la logique de description. Un autre avantage est que les définitions de situations sont intégrées à l'ontologie pour conserver un vocabulaire commun. En outre, SWRL permet de définir des fonctions personnalisées pour enrichir les relations sémantiques entre les éléments composant la définition de situations.

La définition d'une situation par SWRL, basée sur le scénario (cf. chapitre 2), peut être comme suit :

DeviceEvent(?d), has_associated_object(?d, door),
takes_place_in(?d, kitchen),state_value(?d, open),
DeviceEvent(?l), has_associated_object(?l, setLights),
takes_place_in(?l, bedroom),state_value(?l, off), temp :after(?l, ?d)
→ current_state(LightsOffOpenMainDoor, detected)

Ici, les événements représentent l'état courant du système. Dans cet exemple la situation est reconnue quand l'habitant éteint la lumière dans la chambre (pour se préparer à aller au lit) après avoir laissé la porte principale ouverte. Les symboles précédés par *?* son des variables, les autres paramètres des classes ou rôles sont des instances de l'ontologie.

4.6 Conclusion

Ce chapitre présente les méthodes que nous avons conçues pour l'inférence des informations contextuelles qui informent la prise de décision. Ces informations sont la localisation et l'activité courante de l'habitant. En plus, nous présentons la méthode pour la reconnaissance de situations, une étape préalable à la prise de décision. Ainsi, quand une situation est reconnue, l'action plus appropriée à exécuter est choisie prenant en compte les informations contextuelles.

Suite à l'état de l'art, nous avons vu qu'il n'existe pas de méthode qui puisse être appliquée à tous les niveaux d'abstraction. Chaque information contextuelle possède des caractéristiques particulières qui implique que certaines méthodes sont plus pertinentes que d'autres. Par exemple, la localisation peut atteindre de bonnes performances avec les capteurs de présence infrarouge. Cependant, la fusion de données provenant des capteurs complémentaires améliore significativement la localisation. La propagation d'informations à travers les réseaux dynamiques apparaît donc efficace et adaptée puisqu'elle considère la validité temporelle des évidences des sources d'information. La reconnaissance d'activité est une tâche plus complexe à cause de l'incertitude qui affecte le processus d'inférence. Nous avons proposé d'utiliser les réseaux logiques de Markov pour résoudre ce problème. Les résultats obtenus sont très favorables par rapport aux autres méthodes de classification telles que le SVM. De plus, le fait d' avoir un modèle logique est très avantageux pour comprendre les modèles appris et pour inclure des connaissances expertes.

Enfin, nous avons proposé l'application des règles SWRL pour la reconnaissance de situations. L'avantage de cette approche est qu'elle possède une base logique formelle solide et qu'elle s'intègre naturellement à la connaissance du domaine de l'ontologies ce qui permet d'utiliser les termes définis comme concepts et relations. Le modèle résultant est très expressif et améliore la maintenance du système grâce à la lisibilité des définitions de situations et aux algorithmes de vérification de cohérence intégrés dans l'ontologie.

CHAPITRE 5

Prise de décision

Ce chapitre est consacré à la description du processus de décision. La prise de décision demande une représentation formelle des connaissances sur l'environnent et sur les informations à partir desquelles les meilleures actions sont choisies pour garantir le confort et la sécurité de l'habitant. Les informations sur la décision sont de « bas niveau » quand elles proviennent directement des capteurs, et de « haut » niveau si elles sont le résultat des modules d'inférence (p. ex. la localisation et l'activité). Il est également nécessaire de concevoir des modèles de représentation pour les situations dans lesquelles une décision doit être prise.

La solution que nous proposons implique la représentation de concepts à travers des ontologies et règles logiques. Les concepts concernant la décision et les situations d'intérêt sont définis à travers des ontologies utilisant la logique de description et un langage de règles pour le web sémantique (SWRL). Cette représentation permet d'utiliser des raisonneurs logiques pour faire des inférences sur le modèle de représentation. De plus, il est possible d'adapter le modèle à d'autres environnements. Dans la dernière étape de la décision, un modèle MLN est employé pour mettre en œuvre une diagramme d'influence — un modèle pour formaliser le processus de décision — afin de tirer parti des représentations logiques mais aussi pour traiter l'incertitude.

Après un rapide état de l'art des termes clés de la prise de décision dans le cadre de notre recherche, et les méthodes les plus utilisées dans le domaine des environnements perceptifs pour la prise de décision, nous exposons la méthode que nous avons conçue fondée sur les diagrammes d'influence et les MLN.

5.1 Définitions

Dans notre cadre de travail, nous avons défini les éléments suivants :

Définition 13 (Source d'information) *Le système contient un ensemble de n variables V qui décrit l'environnement. Une source d'information est une variable $V_i \in V$, où $i \in \{1...n\}$, avec domaine $Dom(V_i)$ qui représente l'information fournie par un capteur ou un processus d'inférence.*

Définition 14 (État du système) *Si Υ est l'ensemble de valeurs possibles de V, un état du système est une assignation $v \in \Upsilon$ telle que $V = \{V_1 = v_1, V_2 = v_2, ..., V_n = v_n\}$*

Une situation est définie comme :

Définition 15 (Situation) *Une situation $S \subset \Upsilon$ est définie par un ensemble de contraintes $C = \{C_1^{k_1}, C_2^{k_2}, ..., C_m^{k_m}\}$, où chaque contrainte $C_i^{k_i}$ établie un ensemble $A_i \subset DOM(V_{k_i})$ pour restreindre la valeur d'une source d'information V_{k_i}. Ainsi, $S = \{v / \forall C_i^{k_i} \in C, v_{k_i} \in A_i\}$*

Par exemple, si nous avons deux sources d'information, V_1 et V_2, qui correspondent à l'état de la porte principale et la localisation de l'habitant, une situation peut être définie par les contraintes C_1^1 et C_2^2 spécifiant les ensembles $A_1 = \{ouvert\}, A_2 = \{bureau, chambre\}$.

Définition 16 (Situation temporelle) *Considérons une séquence temporelle d'états du système $\delta = (v^{t_1}, v^{t_2}, ..., v^{t_n})$ où t_i est la date d'occurrence. Une situation temporelle R, est définie par un ensemble de restrictions $T = \{T_1, T_2, ..., T_m\}$, où chaque T_k définit une paire de situations (S_k^1, S_k^2) et un intervalle $[a_k, b_k]$ tel que $R = \{(v^{t_i}, v^{t_j}) / \forall T_k \in T, v^{t_i}, v^{t_j} \in \delta, v^{t_i} \in S_k^1, v^{t_j} \in S_k^2, a_k \leqslant t_j - t_i \leqslant b_k\}$*

Ainsi, si une contrainte temporelle T_1 établie un intervalle $[t_i, t_j]$, une situation temporelle sera reconnue quand deux instances de la situation S^1 et S^2 arrivent avec une différence de temps dans l'intervalle. Nous allons nous référer aux situations temporelles simplement comme situations.

Dans le cadre de la prise de décision, le contexte est défini comme suit :

Définition 17 (Contexte) *Ensemble κ des informations qui caractérisent la circonstance dans laquelle la décision est prise.*

Le principal usage du contexte est la désambiguïsation. Quand une situation est reconnue, le contexte fournie l'information complémentaire pour évaluer la circonstance en termes d'une certaine qualité $Q \in \{confort, sécurité, ...\}$. Considérons une fonction F_Q assignant une valeur, dans le domaine de Q, à une situation S. La valeur finale de F_Q dépend de l'information contenue dans le contexte κ : $F_Q(S|\kappa)$. Cette fonction dans notre travail est donnée par le modèle de décision.

Deux définitions essentielles à considérer dans la prise de décision sous incertitude sont le risque et l'utilité :

Définition 18 (Risque) *La probabilité que l'action choisie ait un résultat indésirable.*

Cette définition est assez générale, mais elle est pertinente dans notre cadre de recherche. Le terme Risque peut varier selon le domaine d'application. Ainsi, dans la planification, le risque est ce qui peut faire dépasser le coût ou la durée d'un programme ; en gestion, le risque est la possibilité que les résultats soient différents de ceux prévus. Quelle que soit l'application, le risque est une conséquence de l'incertitude. Pour l'évaluer, il faut lister l'ensemble des possibles sorties indésirables du système, calculer leur probabilité d'occurrence et leurs conséquences. L'évaluation précise du risque peut se faire uniquement en analysant le contexte courant.

Définition 19 (Utilité) *Mesure quantitative qui estime la désirabilité d'un état du système suite à l'effet d'une action prise lors de la prise de décision.*

Une valeur d'utilité doit être donnée à chaque état possible du système. Si l'état correspond à une situation indésirable l'utilité prend une valeur négative. Il existe donc une relation entre le risque et l'utilité : pour calculer le risque d'une action il faut considérer la probabilité de tous les états indésirables, c-à-d. les états ayant une utilité négative.

Un exemple basé sur le scénario 1 décrit dans la section 3.1, dans lequel l'habitant se réveille au milieu de la nuit, permet d'illustrer ces concepts. Dans cet exemple, la situation qui déclenche la prise de décision est la reconnaissance d'un ordre vocal demandant d'allumer la lumière. Les informations qui caractérisent la circonstance et influencent le choix de l'action à mettre en œuvre, le contexte, est composé de la localisation (la chambre), l'heure (minuit), et de l'activité de l'habitant (se reposer). Le choix des lumières à allumer devrait se faire entre celle de plafond, celle de la table de chevet, ou les deux. Les effets des actions, du point de vue du confort de l'habitant, pourraient être : inconfortable, peu confortable et confortable. Ainsi, le risque de chaque action est donné par sa probabilité d'avoir un effet non optimal, en ce cas inconfortable ou peu confortable. L'utilité est une valeur numérique associée à chaque degré du confort de l'habitant.

Il faut noter que le choix dépend du contexte. Dans l'exemple, la meilleure lumière est celle du chevet pour ne pas affecter la sensibilité de la personne étant donné qu'elle vient de se réveiller. Si l'activité en cours était différente, on aurait un contexte diffèrent, et par conséquence un autre critère pour juger la meilleure action à exécuter.

5.2 Méthodes classiques de prise de décision

Il existe dans la littérature de nombreuses méthodes pour la prise de décision qui peuvent être appliquées aux environnements perceptifs. Cependant, la pertinence d'une méthode spécifique dépend des caractéristiques du problème de décision. Dans cette section, nous présentons une brève description des méthodes de prise de décision pertinentes dans le cadre de l'intelligence ambiante.

5.2.1 Programmation linéaire multicritère

Certains problèmes de décision peuvent se modéliser sous la forme de problèmes d'optimisation sous contraintes. L'optimisation multicritère (Steuer, 1986) est une branche de l'optimisation combinatoire dont la particularité est de chercher à optimiser simultanément plusieurs objectifs. La programmation multicritère diffère de la programmation unicritère dans le fait qu'il n'y a pas une seule solution optimale puisqu'il n'est pas possible que tous les critères du modèle soient satisfaits en même temps : l'amélioration de l'un se fait en général au détriment d'au moins un autre critère : par exemple, comment augmenter la vitesse d'une voiture tout en diminuant la consommation de carburant ? Ces méthodes sont

des outils pour aider le décideur à dégager des solutions qui constituent un bon compromis entre les critères conflictuels. Donc la solution finale ne dépendra pas seulement de la méthode choisie mais aussi des préférences du décideur.

Un problème de programmation linéaire multicritère (PLM) prenant en compte K fonctions objectifs linéaires $z_k(x)$, n variables de décision x, et m contraintes, peut s'écrire de la manière suivante :

$$PLM = \begin{cases} \min z_k(x) = c_k \cdot x \ , \ \ k = 1, ..., n \\ x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in D = \{x : x \in \mathbb{R}^n \mid Tx \leq d, x \geq 0\} \\ c_k : (1 \times n), \forall k \, ; \, T : (m \times n) \, ; \, x : (n \times 1) \, ; \, d : (m \times 1) \end{cases} \tag{5.1}$$

sachant que :
- $D \in \mathbb{R}^n$désigne l'ensemble des solutions admissibles,
- chaque composant du vecteur x correspond à une variable de décision,
- les fonctions objectif $z_k(x)$ sont linéaires et sont de la forme $c_1 \cdot x_1 + c_2 \cdot x_2 + ... + c_n \cdot x_n$,
Les restrictions linéaires sur le vecteur x peuvent s'écrire de la façon suivante en explicitant la matrice $T = \{a_{ij}\}$ et le vecteur $d = \{b_i\}$:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 ... a_{1n}x_n < b_1 \\ a_{21}x_1 + a_{22}x_2 ... a_{2n}x_n < b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 ... a_{nn}x_n < b_n \end{cases} \tag{5.2}$$

La résolution d'un tel problème consiste à déterminer un ou plusieurs bons compromis, c'est-à-dire des solutions $x \in D$ ayant la préférence globale du décideur. En plus de l'espace de décision $\mathbb{R}^n$, il est particulièrement utile de considérer l'espace des critères $\mathbb{R}^K$ qui permet de représenter par un point $z(x) = (z_1(x), ..., z_K(x))$ les évaluations d'une solution x. Nous noterons par Z_D l'ensemble des points de $\mathbb{R}^K$ des images des solutions admissibles de D. Un point $z \in Z_D$ est non dominé s'il n'existe aucun point $z', z' \in Z_D$ qui soit meilleur. Une solution $x \in D$ est dite efficace-ou Pareto optimale, si le point $z(x)$ est non dominé.
L'ensemble de solutions efficaces est l'ensemble des solutions intéressantes du point de vue multicritère. Il est cependant généralement de grande cardinalité et souvent infini non dénombrable même dans le cas d'un problème à variables discrètes.
L'objectif du décideur confronté à un problème multicritère est de déterminer une ou plusieurs solutions efficaces qu'il estime satisfaisantes. On peut distinguer trois familles d'approche selon le moment ou le décideur intervient dans le processus de sélection du compromis :

– si les préférences du décideur sont connues préalablement au modèle, on parle d'artic-

ulation *a priori* de préférences. Sous cette approche, il est possible d'utiliser les préférences du décideur pour donner des poids aux fonctions objectifs et transformer le problème multicritère en un problème unicritère qui peut alors être résolu avec la méthode du Simplex ;

– si en revanche, le choix du décideur s'exprime après que le modèle lui a fourni l'ensemble de solutions efficaces, on évoque une articulation a posteriori des préférences ;
– si ces préférences sont exprimées progressivement, au fur et à mesure de l'avancement de la méthode et de la meilleure connaissance de l'interaction entre les critères, il s'agit d'une articulation progressive des préférences.

Il existe de nombreuses méthodes interactives pour trouver des solutions satisfaisantes (Benayoun et coll., 1971; Steuer, 1986).

La décision dans le domaine des maisons intelligentes est souvent affectée par l'existence de plusieurs critères de décision qui sont en conflit. Par exemple, la minimisation des ressources énergétiques et la maximisation du confort des utilisateurs (Tawil et Hagras, 2005). En cela l'optimisation multicritère semble intéressante pour trouver les configurations optimales des variables de décision lors du contrôle domotique d'une maison intelligente. Cependant, cette méthode ne considère pas deux aspects importants des environnements perceptifs : l'incertitude dont le risque est une conséquence, et les relations des dépendances entre les variables de décision. En fait, sous cette approche, on considère que les fonctions à optimiser dépendent des variables mais elles sont indépendantes l'une de l'autre. D'ailleurs, l'optimisation multicritère est bien adaptée à des problèmes ayant un grand nombre de variables et fonctions d'optimisation ; par contre, dans le cadre de nos recherches c'est la prise de décision sous incertitude qui reste le problème majeur plutôt que le grand nombre de variables.

5.2.2 Systèmes experts

Un système expert est un système informatique qui émule l'habilité d'un expert humain pour prendre une décision Bonarini et Maniezzo (1991). La prise de décision est généralement faite par un raisonnement sur une base de connaissances (souvent des règles logiques) plutôt que de suivre une procédure préalablement établie comme c'est le cas de la programmation impérative. Les deux composants principaux d'un système expert sont le moteur d'inférence qui implémente les mécanismes de raisonnement et la base de connaissances qui est dynamique. La connaissance d'un système expert est un ensemble de règles de production ayant la forme *SI...ALORS...*. L'entrée du système consiste en une liste de faits sur l'état du problème et la sortie est une procédure à suivre pour atteindre une solution. Le diagnostic est l'application la plus exploitée de ces systèmes qui ont plusieurs décennies d'utilisation dans les domaines de la médecine, finances, processus de contrôle...

Une façon de traiter l'incertitude dans les systèmes experts est d'ajouter une valeur numérique aux règles (O'Leary et Kandelin, 1988). Chaque valeur exprime la confiance dans la validité de la règle associée. Elle est fixée par un expert du domaine. L'exemple le plus connu est

Mycin (Buchanan et Shortliffe, 1985), un système conçu pour aider les médecins à traiter des infections par bactéries, le modèle introduit le *facteur de certitude*, une valeur réelle entre -1 et 1 pour mesurer la confiance dans les règles logiques.

Les actions de la prise de décision comme trouver des alternatives, les évaluer, et les comparer pour offrir une solution optimale peuvent être faites intégralement par le système ou avec un échange d'information avec l'utilisateur. En effet, un aspect important des systèmes experts est leur capacité à prendre des décisions en interaction avec l'utilisateur (Jr., 1989). Cependant, l'interaction n'est pas abordée dans le cadre de nos recherches, nous considérons que tout le processus de décision est couvert par le contrôleur intelligent.

Les avantages de cette approche sont surtout l'expressivité du modèle, grâce à l'utilisation de règles logiques, et la possibilité de traiter l'incertitude si des dégrées de confiance sont données aux règles. Elle ressemble sur plusieurs aspects à l'approche que nous proposons pour la pris de décision détaillée dans ce chapitre. Pourtant, dans les systèmes experts pour la prise de décision des éléments essentiels comme le risque et l'utilité ne sont pas explicitement définis.

5.2.3 Arbres de décision

Un arbre de décision (Spector, 1993) est un modèle basé sur un graphe en forme d'arbre qui trouve ses origines dans les travaux de Neumann et Morgenstern (1944). Les avantages de ce modèle sont la représentation des variables incertaines, les conséquences des alternatives choisies, l'utilité des alternatives, et le coût de mettre en place chaque alternative. De plus, étant un modèle graphique, la lisibilité est une caractéristique précieuse car elle permet de mieux comprendre les résultats obtenus dans le processus de décision. Un arbre de décision comporte trois types de nœuds :

- (a) les nœuds de décision, représentés par un carré, parmi lesquels nous avons la racine de l'arbre,
- (b) les nœuds aléatoires, qui ne peuvent pas être contrôlés par le décideur car leurs états sont incertains, représentés par un rond, dont leur paramètres sont des probabilités,
- (c) et les nœuds feuilles, représentés par des triangles, qui correspondent aux conséquences des décisions.

Les branches sortant d'un nœud de décision représentent les alternatives disponibles, celles sortant d'un nœud aléatoire représentent des événements incontrôlables. Dans chaque nœud de aléatoire, une probabilité conditionnelle est donnée à chaque branche qui représente la probabilité de l'événement conditionné sur la connaissance disponible dans le nœud. Les feuilles représentent les résultats des décisions plus les éventuelles sorties des nœuds aléatoires qui se trouvent sur le chemin de la racine aux feuilles. L'objectif de trouver la solution optimale – le meilleur ensemble de choix dans les nœuds de décision – est trouvé par un processus récursif (Raiffa, 1970). À partir de chaque feuille en allant vers le nœud racine, on étiquette chaque nœud avec la valeur de la situation qui représente l'espérance de son util-

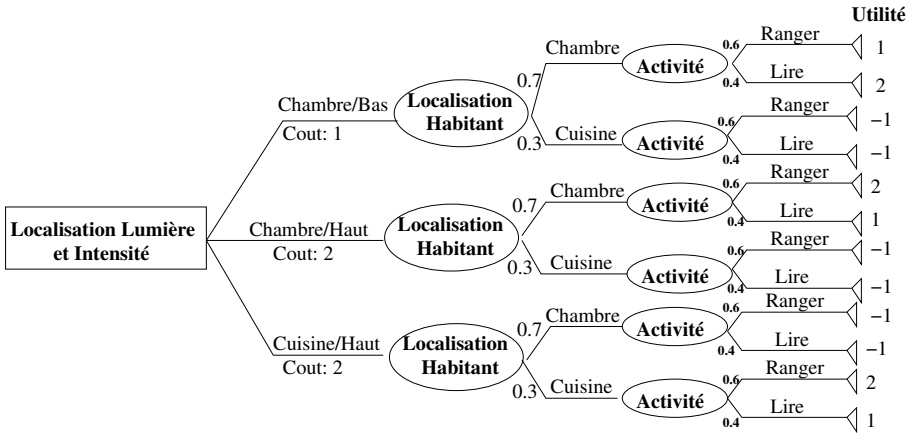


FIGURE 5.1: Arbre de décision pour la reconnaissance d'un ordre vocal.

ité. Dans chaque alternative de décision, il faut soustraire à l'espérance de l'utilité le coût de son exécution. Chaque nœud aléatoire est étiqueté avec la valeur estimée de son successeur et chaque nœud de décision est étiqueté avec la valeur du choix ayant la meilleure utilité.

Dans l'exemple de la figure 5.1, la décision consiste à choisir une lumière dans la maison intelligente et une intensité, après avoir reçu une commande vocale « allume la lumière ». A cet instant, la localisation de l'habitant et l'activité qu'il réalise sont incertains. Les probables valeurs de ces variables et ces probabilités sont cuisine (0.3) et chambre (0.7) pour la localisation, Ranger (0.6) et Lire (0.4) pour l'activité. Dans l'exemple les valeurs d'utilité ont été choisies arbitrairement pour exprimer les préférences du décideur. L'espérance de l'utilité pour l'alternative « chambre/bas » est 0,7*(1*(0,6)+2*(0,4))+0,3*(1*(0,6)+1*(0,4)) - 1 = 0,28. Il est nécessaire d'estimer ainsi les autres alternatives pour prendre la décision optimale.

Cette méthode est très attirante car elle considère non seulement l'incertitude des variables mais aussi caractéristiques pertinents de la théorie de décision pour mieux modéliser la relation causale entre les actions du contrôleur et les objectifs d'améliorer le confort et la sécurité. Un point négatif de cette approche est que la complexité d'un arbre de décision augmente exponentiellement avec le nombre de variables considérés dans le modèle.

5.3 Diagrammes d'influence

Le diagramme d'influence (Howard et Matheson, 1981) est une approche probabiliste pour modéliser des problèmes de prise de décision. Ces diagrammes se basent sur les réseaux bayésiens auxquels des nœuds représentant les actions et les utilités sont ajoutés. Comme les arbres de décision, cette méthode offre une représentation graphique et mathématique d'une situation de décision avec un décideur unique et sous incertitude.

Dans l'approche des diagrammes d'influence, l'incertitude et les conflits de buts (améliorer le confort en économisant d'énergie, par exemple) sont les éléments à partir desquels des actions à suivre doivent être choisies pour arriver dans un état qui lui permette de trouver l'utilité maximale. Si S est l'ensemble des états possibles résultants d'une prise de décision, l'utilité associée à un certain état $s_i \in S$ est donnée par la fonction $U(s_i)$. Si A est un ensemble

de variables d'action et $a \in \Lambda$ une assignation à ces variables où Λ est l'espace de configurations de a. Sous incertitude, le choix a peut entraîner un résultat s_i avec une probabilité $P(s_i \mid A = a)$. Si nous considérons E les faits connus par le décideur au moment de prendre sa décision, nous pouvons alors définir *l'utilité espérée* par l'équation suivante :

$$EU(a \mid E) = \sum_{sinS} P(s \mid A = a, E) \, . \, U(s) \tag{5.3}$$

La prise de décision consiste à évaluer, *a priori*, toutes les actions possibles que l'agent peut exécuter, et choisir celle qui maximise l'utilité espérée. Ainsi, l'algorithme le plus basique pour évaluer le diagramme d'influence est comme suit (Russell et Norvig, 2003) :

input : Structure du diagramme
input : Tableau de probabilité conditionnelle
output: Action optimale

Fixer la valeur des évidences E;
for $a_i \in \Lambda$ **do**
    Fixer $A = a_i$;
    Calculer la probabilité a posteriori des nœuds dans le réseau;
    Calculer $EU(a_i \mid E)$;
end
Choisir l'assignation ayant la meilleure utilité;

D'autres algorithmes ont été proposés dans la littérature pour améliorer la performance en diminuant le nombre de nœuds qui sont évalués lors du calcul de EU (Shachter, 1986; Zhang et Poole, 1992). La figure 5.2 donne un exemple de diagramme d'influence, basé sur le scénario 1 du chapitre 3, où une décision est prise pour répondre à un ordre vocal *«Allume la lumière»*. Dans ce cas, l'assignation de valeurs aux variables d'action— représentées par des nœuds rectangulaires — correspond au choix d'un luminaire à utiliser et à l'intensité de la lumière. Les nœuds ovales sont les variables affectées par les décisions—ils sont appelés nœuds d'état. Les variables qui donnent le contexte pour la prise de décision sont entourées par un rectangle en pointillés. Dans l'exemple l'utilité finale dépend de deux variables d'utilité : le confort et la localisation de la lumière — qui devrait être la même de l'habitant. Cependant, cette localisation peut être difficile à déterminer si l'habitant prononce l'ordre pendant qu'il se déplace.

L'intérêt des diagrammes d'influence est essentiellement leur capacité à exprimer facilement la structure d'un problème de décision et les dépendances et indépendances entre les variables. Graphiquement un diagramme d'influence est une représentation plus compacte qu'un arbre de décision ; par conséquent, il offre une meilleure lisibilité pour modéliser des problèmes complexes.

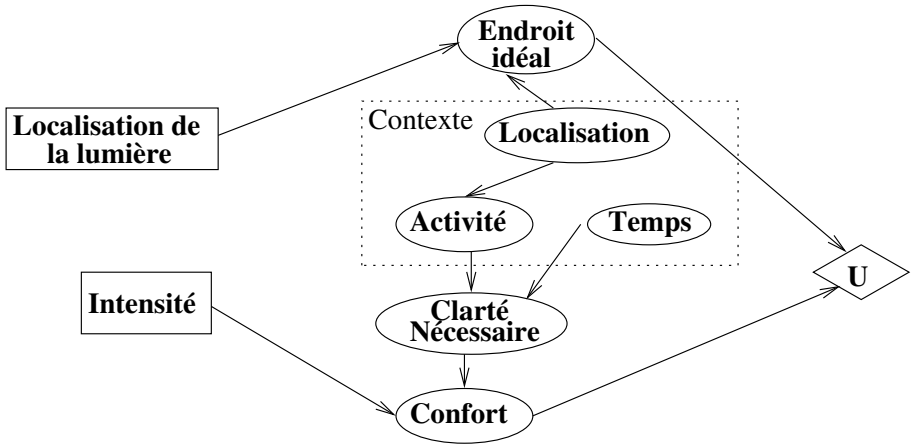


FIGURE 5.2: Diagramme d'influence pour une décision après la reconnaissance d'un ordre vocal.

5.4 Modélisation d'un diagramme d'influence par MLN

Étant donné que les réseaux de Markov sont des modèles plus généraux que les réseaux bayésiens, un diagramme d'influence peut être représenté par un MLN. L'implémentation de la théorie de décision à travers les MLN a donné naissance au Réseau logique de décision de Markov (Markov Logic Decision Network, MLDN) (Nath et Domingos, 2009). Ce modèle permet aux formules logiques du modèle MLN d'avoir des valeurs d'utilité en plus des poids. Ainsi Nath et Domingos (2009) font la définition suivant :

Définition 20 (Réseau de décision de Markov logique) *Un MLDN $M_{L,C}$ est composé d'un ensemble L de triples (F_i, w_i, u_i), où F_i est une formule en logique de premier ordre, w_i et u_i sont des valeurs réelles ; et un ensemble fini de constantes $C = \{c_1, c_2, ..., c_{|C|}\}$.*

Les poids w_i ont le même rôle que dans les MLN. Ils indiquent la confiance dans les formules associés. Les valeurs u_i indiquent la valeur d'utilité obtenue dans le cas où les formules logiques associées sont vraies. Les actions dans le MLDN sont définies par des formules atomiques (ayant un seul prédicat) dans lesquelles la valeur u_i n'indique pas une utilité mais le coût de mettre en place l'action. Il est important de remarquer que l'assignation d'une valeur u_i n'est pas obligatoire pour toutes les formules du modèle car il peut exister des formules qui ne sont pas des actions ou n'interviennent pas directement dans le calcul de l'utilité. Un choix dans le processus de décision consiste à assigner une valeur de vérité à tous les prédicats d'action instanciés.

Le processus de résolution d'un réseau de décision implique la recherche d'une configuration optimale dans l'espace des choix d'actions. Chaque évaluation d'une configuration est une inférence probabiliste qui peut être faite par un algorithme qui calcule la probabilité marginale dans le modèle graphique (le réseau de Markov avec les prédicats instanciés), tel que l'algorithme *Belief Propagation* (Pearl, 1982). La solution la plus simple consiste à itérer sur toutes les configurations possibles jusqu'à trouver celle qui fournit l'utilité maximale. Pourtant, si le problème reste presque inaltéré entre plusieurs itérations, ça peut être

le cas que le niveau de croyance dans plusieurs nœuds (variables) ne sont pas altérés par les changements dans la configurations des nœuds d'action. Il serait avantageux de réutiliser les calculs entre des itérations successives.

Nath et Domingos (2010) ont proposé l'algorithme *Expanding Frontier Belief Propagation* qui met seulement à jour le niveau de croyance des variables significativement altérées entre deux itérations. Ainsi, cet algorithme estime la meilleure assignation d'actions sans effectuer tout le processus d'inférence pour tous les choix, ce qui donne une façon efficace de trouver l'assignation optimale.

Un MLN pour le diagramme d'influence de la figure 5.2 peut être défini ainsi :

Variables	**Domaine**
int	{bas,moyen,haut }
loc	{chambre,cuisine,toilette... }
act	{dormir,cuisiner,ranger,... }
deg	{bon,mauvais,acceptable }

Type	**Predicate**
Action	LocalisationLumiere, Intensite
État	ClarteRequise, Localisation, Activite, Proche
Utilité	EndroitIdeal, Confort

Poids	**Règle**
2,0	$LocalisationLumiere(loc) \wedge Localisation(loc)$ $\rightarrow EndroitIdeal(bon)$
1,8	$LocalisationLumiere(loc1) \wedge Localisation(loc2) \wedge Proche(loc1, loc2)$ $\rightarrow EndroitIdeal(acceptable)$
2,0	$Intensite(int) \wedge Activite(act) \wedge ClarteRequise(act, int)$ $\rightarrow Confort(haut)$
1,2	$Intensite(int1) \wedge Activite(act) \wedge ClarteRequise(act, int2) \wedge int1! = int2$ $\rightarrow Confort(bas)$

Valeurs d'utilité

U(EndroitIdeal(mauvais))=-1	U(EndroitIdeal(acceptable))=0	U(EndroitIdeal(bon))=1
U(Confort(bas))=-1	U(Confort(moyen))=0	U(Confort(haut))=1

'Données d'entrées(Quand elles ne sont pas factuelles, leur probabilité est indiquée)

Localisation(chambre)[0,8]	Localisation(cuisine)[0,15]	Localisation(toilette)[0,05]
Activite(dormir)[0,75]	Activite(lire)[0,17]	Activite(ranger)[0,08]
ClarteNecessaire(dormir,bas)[0,95]	ClarteNecessaire(lire,bas)[0,80]	
Proche(cuisine,chambre)	Proche(chambre,toilette)	

Après que l'ordre vocal soit énoncé, les valeurs des variables de contexte sont obtenues par le module de décision, le modèle de décision est construit en utilisant les règles logiques, et l'inférence est effectuée avec le MLN. Une fois que les valeurs des variables d'action sont fixées, l'inférence donne la probabilité de chaque combinaison de valeurs pour les variables d'utilité, $Confort$ et $EndroitIdeal$. Considérons le cas où les nœuds d'actions sont fixés comme : $a = (LocalisationLumiere(cuisine), Intesite(bas))$, donc pour cette configuration l'utilité espérée est donnée par :

$$EU(a) = \sum_{x \in \{bon, acceptable, mauvais\}} P(EndroitIdeal(x) \mid a).U(EndroitIdeal(x)) + \sum_{x \in \{bas, moyen, haut\}} P(Confort(x) \mid a).U(Confort(x))$$

Outre la simplicité et la capacité de représentation des diagrammes d'influence, nous avons considéré cette approche pour implémenter la prise de décision principalement pour deux raisons : 1) les MLN sont composés de règles logiques, lesquelles peuvent être liées à une ontologie utilisant les concepts déjà définis afin de garder un vocabulaire commun et améliorer la lisibilité du modèle de décision ; et 2) les MLN peuvent traiter l'incertitude associée aux variables de contexte et évidence.

5.4.1 Prise en compte de l'incertitude de l'évidence

L'évidence d'entrée pour le modèle de décision sont toutes les variables du type *État*. Parmi ces variables, quelques unes sont des évidences factuelles. Par contre les variables contextuelles, comme la $Localisation$ et $Activite$ sont incertaines. Elles sont le résultat des autres modules d'inference, et elles sont accompagnées d'une probabilité qui indique un dégré de certitude. Nous avons vu lors de la section 4.3.2 que les valeurs associées aux nœuds contextes dans le réseau dynamique peuvent être normalisées afin d'obtenir un degrée de certitude pour chaque localisation. De même, la méthode de reconnaissance d'activité donne une probabilité à chaque activité.

D'après la règle de Jeffrey (Chan et Darwiche, 2005), pour éstimer la probabilité $P(A \mid B)$ quand la valeur de verité de B est incertaine, l'equivalence suivent peut être utilisée :

$$P'(A) = P(A \mid B) = P(A \wedge B) + P(A \wedge \neg B) = P(A \mid B).P(B) + P(A \mid \neg B).P(\neg B) \tag{5.4}$$

La généralisation dans le cas où l'évidence concerne un ensemble exhaustive d'événements mutuellement exclusifs, $B_1, ..., B_n$, est donnée par la relation :

$$P'(A) = \sum_{i=1}^{n} P(A \mid B_i).P(B_i) \tag{5.5}$$

Pour estimer l'utilité espérée(5.3), il nous suffit de calculer la probabilité des variables d'utilité. Si nous considérons A l'ensemble de variables d'utilité, B les variables contextuelles, et E les évidences factuelles, alors pour estimer $P(A)$ étant donné un B incertaine, il est possible d'utiliser l'équation suivante :

$$P'(A) = P(A \mid B \wedge E).P(B \mid E).P(E) + P(A \mid \neg B \wedge E).P(\neg B \mid E).P(E) \tag{5.6}$$

Étant donné que $P(E) = 1$ l'equation 5.5 est équivalente à l'équation 5.4. Cela nous permet d'utiliser la généralisation 5.5, ayant comme ensemble B toutes les combinaisons pos-

sibles des informations contextuelles, pour trouver la probabilité de chaque configuration possible des variables de utilité (p. ex. $P(EndroitIdeal = bon \wedge Confort = haut)$, $P(EndroitIdeal = bon \wedge Confort = bas)$, ...). Finalement, il suffit de multiplier ces probabilités par les valeurs d'utilité pour obtenir l'utilité espérée.

5.5 Synthèse

Dans ce chapitre nous avons défini notre cadre de travail de prise de décision. Un rapide état de l'art montre qu'il existe un grand nombre de méthodes dans l'état de l'art selon le type de problème de décision. Il peut s'agir d'une décision où interviennent une interaction avec l'utilisateur, le système de prise de décision peut être complètement en charge du processus, la décision peut être multicritère ou monocritère, le problème peut faire intervenir des variables stochastiques. Dans le cadre de nos recherches, les diagrammes d'influence est l'approche la plus pertinente car il couvre les caractéristiques désirées de notre modèle de prise de décision : une représentation graphique claire des variables de décision et leurs relations, et la capacité de traiter l'incertitude à travers les probabilités. Ce chapitre se termine avec la méthode que nous avons implémentée, les réseaux logiques de décision de Markov, une extension des MLN qui permette d'implémenter des diagrammes d'influence comme un ensemble de règles logiques pondérées.

CHAPITRE 6

Expérimentations et résultats

Dans ce chapitre, nous présentons les expérimentations effectuées dans un environnement perceptif pour d'une part acquérir des données permettant de valider les méthodes d'inférence et de prise de décision et, pour d'autre part tester le fonctionnement du système SWEET-HOME. Les méthodes d'inférence des informations de haut niveau et de prise de décision ayant été introduites dans les sections précédentes ainsi que la présentation des fonctionnalités du contrôleur intelligent, nous allons nous concentrer sur la présentation des expérimentations et des résultats.

Le but des expérimentations qui ont été menées était d'obtenir des résultats dans l'analyse des informations contextuelles et la prise de décision concernant l'habitat intelligent dans des conditions les plus proches possibles de la réalité. Pour ce faire, nous avons réalisé deux expérimentations dans le même environnement perceptif qui ont visé plusieurs objectifs :

1. Ces expérimentations ont permis de **collecter des données** pour permettre l'acquisition de modèles d'inférence. Les modèles d'inférences nécessitent un ensemble de données important pour pouvoir calculer les statistiques essentielles à l'apprentissage de modèles probabilistes. Étant donné le manque de données enregistrées dans les habitats intelligents et librement disponibles, comme cela a été établi par Fleury et coll. (2012)), cette étape d'expérimentation est essentielle.

2. L'autre apport essentiel des expérimentations est de fournir des situations réalistes permettant la validation et l'**évaluation** des progrès qui ont dû être effectués tout au long du travail de thèse.

3. Enfin, ces expérimentations ont permis d'effectuer des **tests fonctionnels** permettant de valider l'architecture logicielle et les choix techniques fait au niveau du projet.

Le chapitre présente l'environnement perceptif de référence pour le projet SWEET-HOME en section 6.1 et les deux expérimentations ainsi que la démarche suivie en sections 6.2 et 6.3. Puis, nous présentons les résultats obtenus par l'application des méthodes de localisation envisagées en section 4.3.2, de reconnaissance d'activité (en section 4.4.2), de reconnaissance de situations (en section 4.5.2), et enfin de prise de décisions (section 5.4).

6.1 Environnement perceptif utilisé pour les évaluations

La plateforme d'expérimentation DOMUS a été conçue et mise en place par l'équipe Multicom du Laboratoire d'Informatique de Grenoble (Gallissot et coll., 2013). Cet appartement fonctionnel de 34 m^2 est équipé de capteurs et d'actionneurs afin d'agir sur l'environnement (éclairage, volets, systèmes de sécurité, chauffage, ventilation, contrôle audio-vidéo ...). DOMUS est consacré à l'observation et l'analyse de l'interaction entre l'utilisateur et les systèmes d'intelligence ambiante. Plus de 150 capteurs, actionneurs et sources d'informations sont gérés dans cet appartement.

L'architecture technique de DOMUS se base sur le système de bus KNX (KoNneX) [1], un standard ISO (ISO/IEC 14543) pour le contrôle domotique dans les bâtiments. Outre KNX, plusieurs systèmes de bus coexistent dans DOMUS, tel que UpnP (*Universal Plug and Play*) pour la gestion audio et vidéo, X2D pour les contacts de porte, fenêtres, placards, etc., RFID pour la détection de l'interaction avec des objets tangibles. Des modifications ont été nécessaires pour son utilisation dans le projet SWEET-HOME, l'appartement a ainsi été équipé de sept microphones sans fil placés dans le faux plafond (2 pour chaque pièce, sauf dans la salle de bains). La figure 6.1 décrit l'emplacement des différents capteurs.

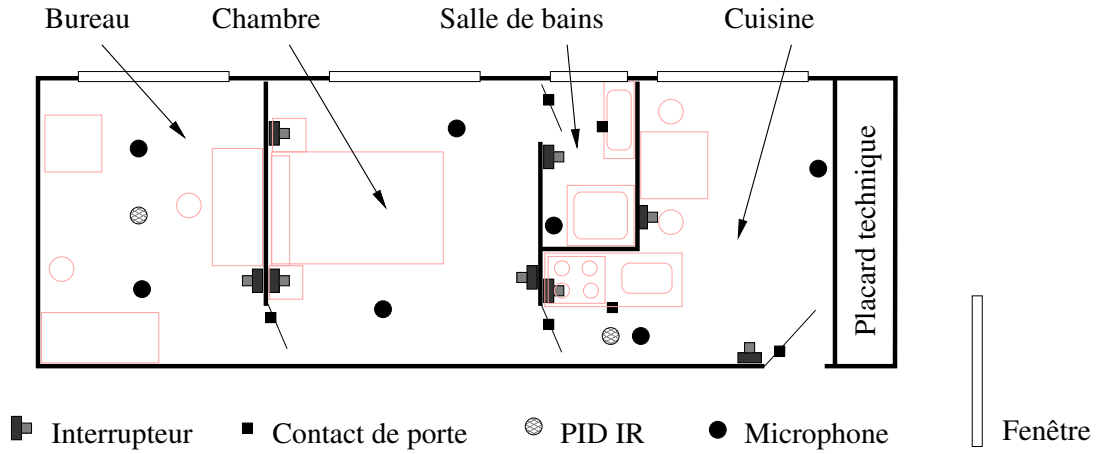


FIGURE 6.1: Plan de l'appartement DOMUS et disposition des capteurs.

Les différents types de capteurs utilisés dans nos travaux de recherche sont les suivants :

- les microphones intégrés au plafond. Ces microphones peuvent être utilisés pour détecter des sons anormaux, retranscrire la parole, ou plus généralement classer les sons et les paroles qui sont enregistrés dans l'appartement ;
- un grand nombre de capteurs sur bus KNX (norme ISO/IEC 14543) permettant d'identifier l'utilisation d'une porte (de communication entre des pièces ou une porte de commodité - réfrigérateurs, placards, etc.), d'une prise électrique, d'un interrupteur pour la lumière, d'un volet, etc.
- des capteurs « environnementaux », avec notamment des capteurs de luminosité, des capteurs de température et d'hygrométrie, ou encore un capteur de CO_2 ;
- un compteur électrique et un compteur d'eau ;

1. www.knx.org

– un ensemble de caméras de surveillance qui servent uniquement à suivre les expérimentations en temps réel et à annoter celles-ci. Elles ont été installées dans toutes les pièces sauf dans la salle de bains par égard pour l'intimité de la personne.

Par ailleurs, l'appartement DOMUS est équipé d'actionneurs qui seront mis en œuvre par le biais des interrupteurs classiques, mais aussi grâce à des ordres envoyés depuis le contrôleur intelligent dans la deuxième expérimentation. Un système e-lio de la société Technosens a été installé, il peut aussi être piloté par le contrôleur pour mettre en relation la personne présente dans l'appartement avec un proche par visioconférence.

Le fait que cette plateforme soit installée comme un appartement classique permet d'organiser des expérimentations dans lesquelles les personnes vont réaliser leurs activités de la vie quotidienne d'une manière que l'on peut estimer sensiblement équivalente à celle qui sera la leur dans leur domicile.

6.2 Première expérimentation : constitution d'un corpus multimodal

6.2.1 Démarche expérimentale

6.2.1.1 Protocole expérimental

Quatre phases d'expérimentation ont été conduites dans le cadre du projet SWEET-HOME, chacune durait généralement un mois. Celles-ci étaient précédées d'une étape de recrutement. Une fois le sujet recruté, il se rendait sur le lieu d'expérimentation où les objectifs lui étaient expliqués. Il devait ensuite signer une feuille de consentement qui spécifiait ses droits et lui expliquait les dispositions prises pour assurer la confidentialité des données issues de l'expérimentation.

Avant chaque expérimentation, le participant était guidé dans l'appartement pour se familiariser avec les différentes pièces, les objets à utiliser, et les interrupteurs utilisés pour agir sur les actionneurs. Grâce à cette accoutumance, le participant pouvait agir ensuite plus naturellement dans l'environnement lors de l'enregistrement.

Puis, une feuille comportant la description d'un scénario détaillé à suivre était fournie au participant pour le guider à tout instant tout au long de l'enregistrement. L'ordre des activités à réaliser a été fixé *a priori*; cependant, le participant était libre de réaliser l'activité à sa façon. La description du scénario incluait une durée de réalisation indicative pour chaque activité, mais le participant pouvait disposer du temps qu'il voulait.

Au moment où le participant pénétrait dans l'appartement, les outils d'enregistrements (ici, serveur et traces KNX, enregistreur vidéo et enregistreur sonore) étaient mis en route, ils étaient arrêté en fin d'expérimentation. La communication entre le participant et l'équipe technique était assurée par une oreillette.

6.2.1.2 Contraintes imposées aux sujets

En générale, la personne était supposée vivre dans l'appartement comme chez elle, sans se préoccuper des caméras ni des capteurs environnants. Les seules contraintes imposées ont été les suivantes :

- Le sujet devait faire les activités selon l'ordre indiqué dans le scénario. En fait, il s'agissait de celui qui rythme habituellement une journée.
- Toutes les activités devaient être réalisées au moins une fois.
- Le sujet devait rester tout seul dans l'appartement pendant le déroulement du scénario.
- Le sujet ne pouvait pas communiquer avec l'extérieur, sauf dans le cas où la communication était spécifiée dans le scénario.

6.2.1.3 Activités à réaliser

Nous avons défini sept activités (se reporter à la section précédente 4.4), chacune composée de plusieurs actions. Par la suite, nous chercherons à les différentier automatiquement à l'intérieur de l'appartement à partir des informations du réseau domotique. Elles permettront également d'évaluer le contexte sur lequel la prise de décision est effectuée.

1. **Préparer et prendre son repas.** Des ingrédients et matériel pour préparer un petit-déjeuner étaient disponibles dans le placard de la cuisine. Le sujet n'avait aucune contrainte par rapport à la préparation de son repas ni sur le temps employé.
2. **Nettoyer et ranger.** Cette activité est effectuée dans la cuisine et la chambre. Dans la première pièce, il s'agissait de laver la vaisselle, nettoyer la table après avoir pris le repas, et ranger la vaisselle dans le placard. Dans la chambre, le nettoyage comporte le rangement du lit après avoir dormi. Le sujet devait aussi passer l'aspirateur dans ces deux pièces.
3. **Activités d'hygiène.** Cette activité est plus complexe à faire. Même s'il y a une douche dans la salle de bains, il serait difficile de convaincre les sujets de prendre une douche dans un appartement équipé des capteurs. On leur a uniquement demandé de se brosser les dents et laisser couler l'eau pour feindre de prendre une douche.
4. **Communication.** Pour cette expérimentation, la personne recevait un appel téléphonique depuis le bureau lorsqu'elle était dans la cuisine, elle devait laisser sonner le téléphone. Elle devait ensuite en simuler une. La personne n'était pas en réelle communication avec l'extérieur, elle disposait d'une liste de phrases qu'elle devait énoncer avec le téléphone en main comme s'il s'agissait d'une communication.
5. **S'habiller et se déshabiller.** Un ensemble de vêtements était disposé dans la commode de la chambre. On demandait seulement au participant de les enfiler par dessus ses vêtements.
6. **Dormir.** Le sujet s'installait sur le lit et se reposait quelques minutes sans durée imposée.

Type d' Annotation	Instant/ Intervalle	Description	Domaine
Localisation	Intervalle	La pièce où la personne se trouve	Chambre, bureau, cuisine, salle de bain
Activité	Intervalle	L'activité en cours de réalisation	Dormir, détente, habillage, repas, hygiène, nettoyage, communication
Action	Intervalle	Une action spécifique dans l'activité	Boire, manger, faire la vaisselle laver, ranger le lit, lire, etc.
Position	Intervalle	La posture du sujet	Debout, allongé, assis
Manipuler	Intervalle	Type d'objet manipulé par le sujet	Ingrédient, aspirateur, ordinateur, textile, papier, vaisselle,etc.
Ouvrir	Instant	Si une porte a été ouverte ou fermée	Réfrigérateur, placard, volets, rideaux, etc.
Son	Instant	Un son produit par l'utilisation d'un objet	Vaisselle, papier, aspirateur,etc.
Parole	Instant	Parole prononcé par le sujet	Transcription de la parole

TABLE 6.1: Types d'annotation

7. **Se détendre.** Le sujet avait la possibilité d'écouter de la musique ou d'utiliser l'ordinateur dans le bureau. D'ailleurs, il pouvait amener de la lecture dans le bureau ou la chambre.

6.2.2 Annotation du corpus recueilli

Trois types d'informations ont été annotés à partir des enregistrements des expérimentations : l'interaction du sujet avec les objets, les énonciations du sujet, et les états dans lesquels il se trouve. L'interaction concerne la manipulation des objets spécifiques (la vaisselle, les livres, les vêtements, etc.) ainsi que l'utilisation d'appareil domestiques tel que le réfrigérateur ou la cafetière. Les énoncés sont les paroles prononcées. Les états sont la localisation de l'utilisateur et les actions qu'il réalise (chacune appartient à une activité définie).

Les types d'annotations que nous avons utilisés pour l'évaluation des expérimentations sont résumés dans le tableau 6.1. Pour l'état des portes, nous annotons uniquement les changements d'état « 1/0 » à un instant précis. Pour les autres annotations, une paire « attribut = valeur », sous forme de texte, a été utilisée car cela donne plus de flexibilité pour inclure d'autres attributs pour la même annotation. Par exemple, un attribut de commentaire peut être ajouté à l'annotation pour exprimer qu'il existe une ambiguïté sur l'action réalisée par le sujet. Les métadonnées des annotations, comme le nom de l'auteur et la date d'annotation, sont sauvegardés automatiquement. Chaque type d'annotation prend toujours une valeur unique (on ne considère pas des activités parallèles ou localisations floues). Une annotation peut décrire l'occurrence d'un événement en un instant précis ou un état dans un intervalle de temps.

Plusieurs outils d'annotations ont été évalués au préalable, tel que Observer XT, ANVIL, ELAN et Advene [2]. Tous ces logiciels offrent des fonctions standard pour annoter des don-

2. http ://liris.cnrs.fr/advene/

nées audiovisuelles, des facilités pour naviguer dans les vidéos, le support de différents formats, l'organisation des annotations en couches. Advene a été choisi parce qu'il peut traiter des vidéos de durée très longue, ce qui n'a pas été possible avec les autres outils. De plus, Advene organise les éléments des annotations — tel que les types d'annotation, les annotations, et leurs relations — selon des schémas qui sont définis par l'utilisateur.

Une image d'Advene avec les vidéos de notre corpus est présentée dans la figure 6.2. Le panneau d'annotation montre les différents types d'annotation (un pour chaque ligne) et leurs instances correspondantes dans la dimension temporelle.

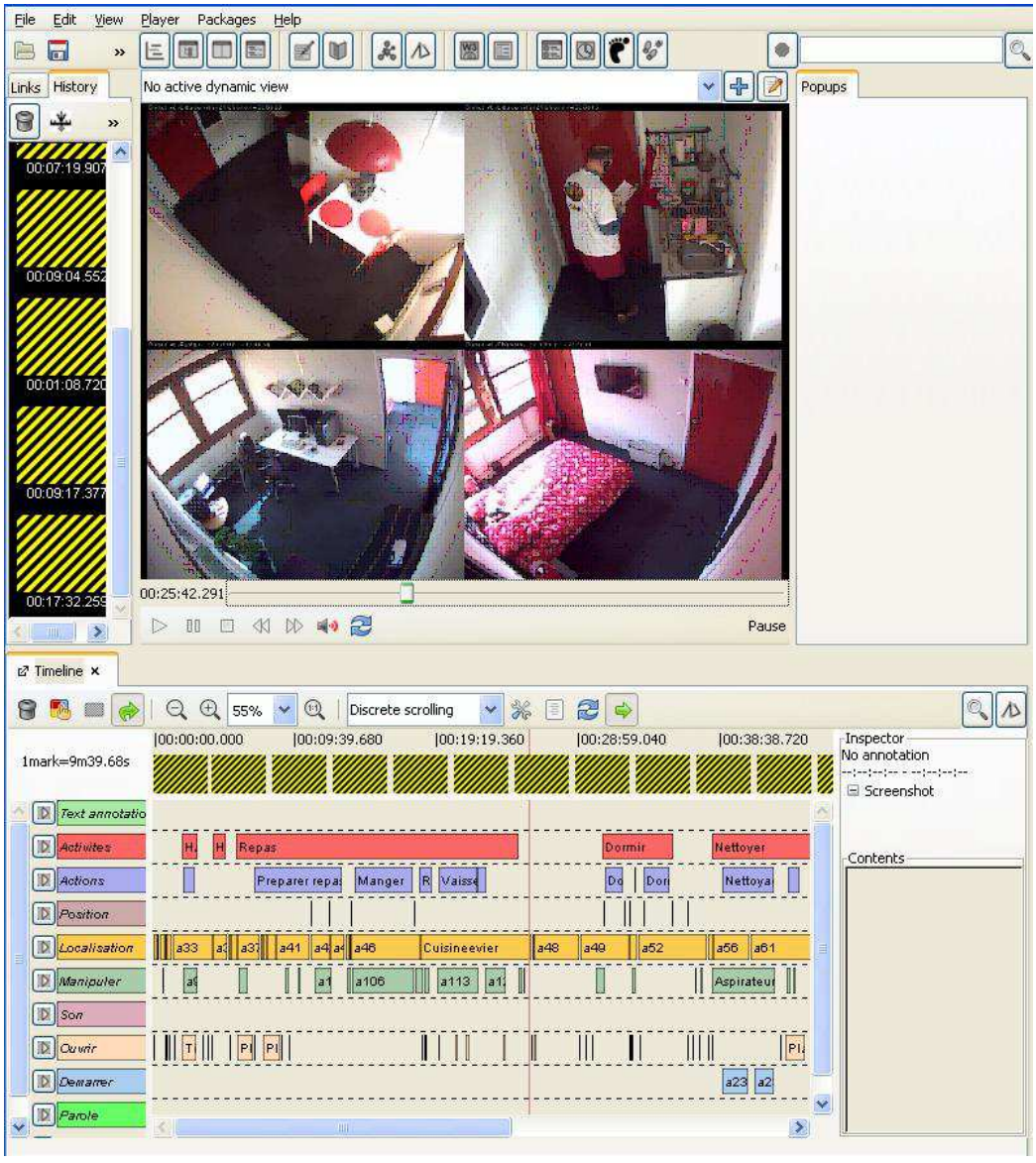


FIGURE 6.2: Exemple d'annotation du corpus avec Advene

6.2.3 Enregistrements

Cette expérimentation a donc principalement permis d'acquérir un corpus multimodal utilisable pour nos recherches concernant la localisation et de la reconnaissance d'activité. À cette fin, 21 personnes ont été recrutées et ont dû jouer, le plus naturellement possible,

des scénarios prédéfinis qui comportaient différentes activités (habillage, hygiène, petit-déjeuner, vaisselle, sortie pour les courses, ménage, écoute de la radio, ...) dont deux conversations téléphoniques. La moyenne d'âge des 21 participants était de 38.5 ans et s'échelonnait entre 22 et 63 ans. Ensuite, les données recueillies dans cette expérimentation ont été utilisées surtout pour apprendre et évaluer les modèles de localisation et reconnaissance d'activité.

Chaque session d'enregistrement a duré environ 2 heures pour une durée totale de plus de 26 heures et 7 minutes. Ces activités ont été annotées avec le logiciel ADVENE. L'annotation a donc été un processus assez lourd et gourmand en temps.

6.3 Deuxième expérimentation : évaluation du contrôleur intelligent

Dans cette deuxième expérimentation, nous avons mis en place un système d'analyse audio temps réel et le contrôleur intelligent pour piloter l'environnement perceptif. Outre l'évaluation du contrôleur, nous nous intéressons à l'analyse des informations concernant l'interaction vocale entre la personne et le contrôleur intelligent, la réponse aux ordres vocaux, la reconnaissance de situations et la prise de décision en contexte.

13 sujets francophones ont participé à l'expérimentation pour réaliser des activités de vie quotidiennes. En fait ces activités sont un sous-ensemble de celles réalisés dans la première expérimentation. Cependant, la commande vocale a permis de prononcer des ordres vocaux pour agir sur les actionneurs. La moyenne d'âge des 13 participants était de 35 ans et s'échelonnait entre 19 et 62 ans.

6.3.1 Démarche expérimentale

Le nombre d'activités a été limité pour réduire le temps d'expérimentation :

1. **Préparer et prendre son repas.**
2. **Nettoyer et ranger.**
3. **Communication.**
4. **Dormir.**
5. **Se détendre.**

La personne devait prononcer un minimum de 15 ordres domotiques au cours de ses activités, il devait les répéter en cas d'insuccès au bout de 4 secondes, et ce jusqu'à 3 fois. Un magicien d'Oz permettait de poursuivre l'expérimentation en cas d'insuccès. Pour certaines actions, le participant était laissé libre d'utiliser un interrupteur ou la commande vocale.

Les ordres domotiques devaient respecter une grammaire présentée dans la section suivante.

Pour cette expérimentation, la personne devait prononcer, à un certain moment du scénario, une commande de détresse ou de mise en relation avec un proche. Le contrôleur la

Entité lexical	**Structure**
basicCmd	key initiateCommand object | key stopCommand object | key emergencyCommand |
key	« Nestor » | « Maison »
stopCommand	« stop » | « arrête »
initiateCommand	« ouvre » | « ferme » | « baisse » | « éteins » | « monte » | « allume » | « descend » | « appelle » | « donne »
emergencyCommand	« au secours » | « à l'aide »
object	[determiner] (dispositif | person | organisation | information)
determiner	« mon » | « ma » | « l' » | « le » | « la » | « les » | « un » | « des » | « du »
dispositif	« lumière » | « store » | « rideau » | « radio »
person	« fille » | « fils » | « femme » | « mari » | « infirmière » | « médecin » | « docteur »
organisation	« samu » | « secours » | « pompiers » | « supérette » | « supermarché »
information	« température » | « heure »

TABLE 6.2: Grammaire des ordres domotiques

mettait alors en contact avec quelqu'un à l'extérieur de l'appartement à travers le système e-lio. Cette communication avec l'extérieur était très courte car juste après la mise en contact, on demandait au participant de finir la communication et continuer le scénario.

Des situations de risque étaient prévue (porte d'entrée ouverte lors de la sieste), le contrôleur devant envoyer un message sonore d'alerte à la personne pour qu'elle corrige la situation.

6.3.2 Grammaire des ordres domotiques

La grammaire définit la structure que doit avoir un ordre domotique pour qu'il soit reconnu par le contrôleur intelligent. Elle est spécifiée dans le tableau 6.2.

Quelques exemples d'ordres domotiques validées par cette grammaire sont : « Nestor baisse le store », « Maison appelle mon fils », « Maison donne l'heure », etc.

6.3.3 Logiciel PATSH

Les signaux audio produits lors de l'expérimentation ont été traités avec le système PATSH, développé dans le cadre du projet SWEET-HOME, dont l'architecture fonctionnelle est décrite dans la figure 6.3. Succinctement, les événements audio sont détectés, en temps réel, et sont ensuite classés comme parole ou comme son de la vie courante (p.ex. le claquement d'une porte). Le résultat de la classification sonore est une entrée du contrôleur.

Le système PATSH se découpe en deux ensembles : capture et chaîne de traitement audio. Ces deux éléments peuvent coexister ou non sur la même unité de traitement. Une couche logicielle d'échange des événements sera en mesure d'abstraire le transfert des données (local, client/serveur) entre ces deux ensembles. Cette modularité offre différentes possibilités d'installation des systèmes au sein d'un appartement et de diversifier notamment les sources

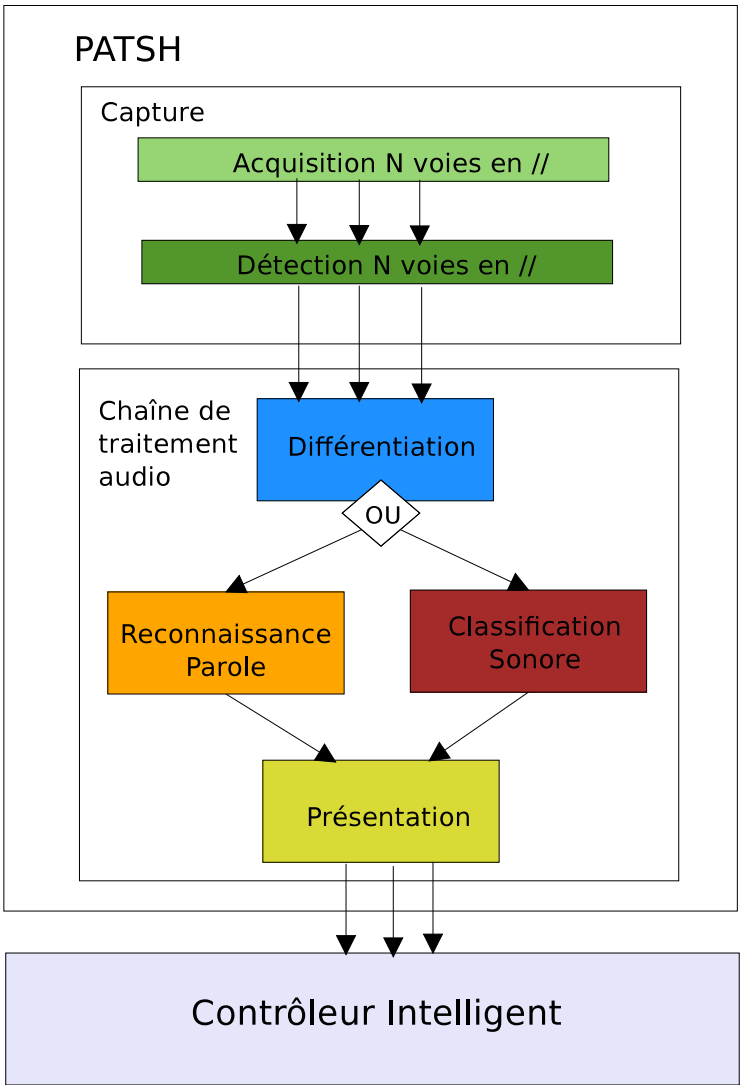


FIGURE 6.3: Modules fonctionnels du système PATSH.

d'acquisition. Cette séparation offre également des perspectives d'adaptation de systèmes existants plus simple et de fait de garantir l'évolutivité du système.

Un système de capture est définit comme étant un couple de modules d'acquisition et de détection d'événements audio. La capture fournit des objets d'événement audio à la suite des modules de traitements. Le système PATSH peut supporter plusieurs systèmes de capture (locaux ou déportés) définis dans sa configuration générale. Une chaîne de traitement audio est déclarée pour chaque source d'acquisition. Chaque source d'acquisition est composée de plusieurs canaux audio dont l'acquisition est synchronisée. Une source d'acquisition peut fournir plusieurs canaux audio synchronisés. Ainsi, un module d'acquisition est écrit pour chacune de ces sources (abstraction de drivers pour l'étage de détection). Les différents modules d'acquisition devront alors présenter des données identiquement structurées à la chaîne de traitement. L'ensemble des systèmes de capture écrit les événements audio dans une pile commune à une destination de l'unité de traitement audio qui les consomme indifféremment/indépendamment.

La chaîne de traitement audio de PATSH regroupe les étages de différentiation, de clas-

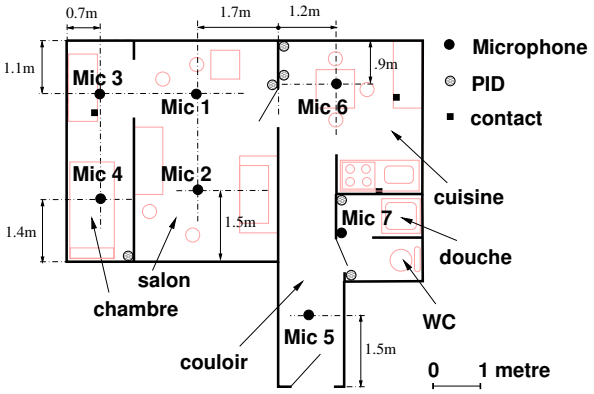


FIGURE 6.4: Plan de l'appartement et position des capteurs.

sification/reconnaissance de parole et de notifications au contrôleur intelligent (module de présentation). Le système PATSH exécute la chaîne de traitement sur chaque événement audio par l'intermédiaire d'un pool de threads (ensemble d'unités de traitement, processus légers, dont le nombre est automatiquement géré par la plateforme en fonction des capacités de calcul du serveur). Chaque événement audio se trouvant dans la pile d'entrée de la chaîne de traitement audio est donc traité de manière indépendante des autres canaux. Une étiquette de temps associé à l'événement permettra alors au contrôleur de traiter les événements simultanés.

6.4 Évaluation de la méthode de localisation

Cette section a pour but d'évaluer la méthode de localisation dans l'appartement présentée à la section 4.3.2. En plus des données issues des expérimentations dans l'appartement DOMUS, des données obtenues dans un autre environnement perceptif ont été utilisées, l'Habitat Intelligent pour la Santé (HIS) de la Faculté de Médecine de Grenoble (Le Bellego et coll., 2006).

La figure 6.4 décrit le HIS. Cet appartement pleinement fonctionnel est composé de 6 pièces et est équipé des capteurs suivants : 7 microphones (Mic) placés dans le plafond (à hauteur d'environ 3 mètres), 3 capteurs de contact sur des portes de meubles (CP), et 6 détecteurs de présence infrarouge placés sur les murs (PID).

L'utilisation de la méthode définie nécessite le calcul de poids nécessaires à la détermination des niveaux d'activation.

6.4.1 Calcul de poids dans le cas du HIS

Le poids w_{o,h^i} a été calculé différemment pour chaque type de capteur. Pour une observation o, avec $o.type \in \{CP, PID\}$, un seul nœud hypothèse de localisation est créé. Les informations spatiales des PID et des CP sont univoques et certaines. Par exemple, l'ouverture du réfrigérateur ne peut se produire que si la personne se trouve dans la pièce. Par conséquent, un événement est lié avec un poids $w_{o,h} = 1$ à une hypothèse unique de locali-

sation, et l'hypothèse est liée avec un poids de $w_{h,c} = 1$ au contexte. Il n'en va pas de même pour les informations des microphones. En effet, les microphones peuvent théoriquement capter toutes les ondes acoustiques générées dans les pièces d'une habitation. L'énergie du signal sonore détecté fournit bien une information dynamique sur la proximité de la source, mais il ne donne par contre aucune indication sur sa direction. Par exemple, un bruit capté à 1 mètre d'un microphone peut tout à fait avoir été généré dans une pièce différente de celle dans laquelle le microphone est placé. Pour prendre en compte ce phénomène, nous avons modélisé l'ambiguïté d'un microphone en fonction de sa distance avec les autres pièces par les poids w_{o,h^i} des relations dans le réseau dynamique. La valeur de ces poids a été calculée en estimant la probabilité $P(loc = i|Mic = j)$ que l'habitant soit à la i^e localisation sachant qu'un événement a été détecté sur le j^e microphone. Le poids de la relation entre l'hypothèse h^i et l'observation o est donné par la formule 6.1 :

$$w_{o^k,h^i}(t_n) = \frac{P(loc = i|o^k_{t_n}.capteur) \times o^k_{t_n}.rsb}{\sum_{l \in L} \sum_{o \in O_{t_n}} P(loc = l|o.capteur) \times o.rsb} \tag{6.1}$$

où O_{t_n} est l'ensemble des observations simultanées provenant des microphones à l'instant t_n, $P(loc|Mic)$ est une connaissance *a priori*, L est l'ensemble des localisations possibles, et $o.rsb$ est le rapport signal sur bruit linéarisé de o. Ainsi, w_{o^k,h^i} est un poids normalisé ($\sum_{i,k} w_{o^k,h^i} = 1$) qui représente le lien entre les observations sonores simultanées et les hypothèses de localisation. Pour calculer la probabilité $P(loc = i|Mic = j)$, deux approches ont été testées, l'approche naïve et l'approche par analyse statistique de corpus. Pour l'approche naïve, la référence sonore a été choisie à 1 mètre au dessous du microphone. À partir de cette référence, un cercle de 2 mètres de rayon a été tracé autour de chaque microphone, ce qui correspond à une atténuation de $-6dB$ en considérant l'atténuation quadratique classique. Au delà de ce cercle, la perte d'énergie est supérieure à 75%. Le poids est calculé selon la surface de l'intersection entre le cercle et les pièces, avec une pénalité de 2 lorsqu'une cloison est traversée.

L'autre approche est statistique ; elle calcule directement les probabilités à partir du corpus annoté. Le tableau 6.3 indique les poids obtenus pour l'approche naïve et pour l'approche statistique.

Le mélange d'informations *a priori* et d'informations dynamiques permet une meilleure désambiguïsation. Par exemple, dans le cas de deux observations sonores simultanées captées par le micro 6 de la cuisine et le micro 7 de la salle de bains avec un RSB identique égal

	$P(Loc\|Mic)$ estimé par la méthode naïve							$P(Loc\|Mic)$ estimé à partir du corpus						
Mic	1	2	3	4	5	6	7	1	2	3	4	5	6	7
chambre	0,14	0,07	0,70	0,85				0,28	0,29	0,42	0,43	0,25	0,18	0,20
salon	0,86	0,93	0,27	0,14	0,01	0,13	0,03	0,59	0,56	0,47	0,41	0,07	0,07	0,09
cuisine			0,03	0,02	0,10	0,87	0,50	0,05	0,08	0,06	0,09	0,45	0,63	0,37
douche					0,06		0,18	0,06	0,05	0,04	0,04	0,09	0,04	0,10
WC					0,06		0,18	0,01	0,01	0,01	0,01	0,12	0,05	0,21
couloir					0,77		0,10		0,02	0,01	0,02	0,03	0,03	0,02

TABLE 6.3: Estimation de $P(Loc|Mic)$ dans le cas de HIS

Capteurs	PID	CD	Mic+CD	PID+CD	PID+Mic	PID+Mic+DC
DOMUS sans a priori	62,9	59,9	63,7	71,6	64,5	**73,2**
DOMUS a priori	62,8	73,3	77,4	81,7	64,6	**84,0**
HIS sans a priori	88,9	26,5	32,8	**89,4**	87,7	88,2
HIS a priori naïf	88,9	26,5	34,1	89,4	89,0	**89,5**
HIS a priori statistique	88,9	26,5	34,8	89,4	89,7	**90,1**

TABLE 6.4: Exactitude(%) avec plusieurs combinaisons de sources

à $12dB$, les formules 4.21 et 6.1 et l'a priori naïf donnent pour l'activation

$$A^{h^{cuisine}} = rsb \times (.87 + .5)/[rsb \times (.87 + .5 + .18 + .13 + .03 + .18 + .1)] = 0,69 \tag{6.2}$$

qui sera supérieure à celle de la salle de bains $A^{h^{sdb}} = 0,09$ bien que le rapport signal sur bruit soit le même.

6.4.2 Calcul de poids dans le cas de DOMUS

w_{o,h^i} pour CP et PID a été calculé de la même façon que pour le HIS. Cependant, les CP des portes qui séparent deux pièces sont ambigus par rapport à la localisation de l'habitant. Le problème est de décider à quelle pièce donner le poids le plus important. Dans ce cas, le poids du lien est défini par l'équation 6.3 par une probabilité conditionnelle

$$w_{o,h^i}(t_n) = P(loc = i|Mic = j) \tag{6.3}$$

où $o.state \in \{Open/Close\}$ et $\mathscr{C}$ est la localisation de l'habitant à l'instant t_{n-1}. Cette probabilité a été apprise sur un corpus différent de celui utilisé dans les expérimentations.

6.4.3 Expérimentations et résultats

Pour chaque enregistrement, les observations enregistrées par les capteurs CP, PID et Mic ont activé un réseau dynamique. Les traces d'activation de chaque contexte ont été conservées. L'emplacement du participant au début de l'expérimentation est connu, ici il s'agit toujours de la cuisine.

Les performances de localisation ont été évaluées à l'échelle de la seconde. À chaque seconde, le contexte de localisation de plus haut poids est comparé à la vérité « terrain ». S'il y a correspondance, alors il s'agit d'un vrai positif (VP), sinon il s'agit d'une confusion. Cela nous permet de générer une table de confusion à partir de laquelle le taux d'exactitude est calculé par $TE = \frac{nb(VP)}{nb(test)}$, où $nb(test)$ correspond à la durée de l'évaluation exprimée en secondes.

Pour les données de DOMUS, une première expérimentation a été faite sans utiliser l'information *a priori* sur les contacts de porte. La méthode a été exécutée une deuxième fois en utilisant ces informations pour évaluer leur contribution. De la même façon, trois expérimentations ont été réalisées avec le corpus de HIS :

1. sans *a priori* ($P(loc \mid Mic) = 1$ quand le microphone est dans la pièce, 0 si non),
2. avec une approche naïve,

3. et enfin avec l'approche statistique.

Le tableau 6.4 montre les résultats des deux corpus en utilisant plusieurs combinaisons de capteurs.

Dans le cas de DOMUS, il est clair que la fusion d'information améliore l'exactitude puisque celle-ci augmente quand plus de capteurs sont utilisés. Même quand la précision des capteurs infrarouges est bonne, les performances ne sont pas très bonnes (63%). La deuxième ligne montre l'exactitude estimée avec l'information statique des capteurs de porte. Dans tous les cas, l'information *a priori* améliore les résultats, ce qui confirme que l'information statique peut être utilisée pour améliorer la performance.

Avec le corpus HIS, il apparaît quelques cas où la fusion d'information n'améliore pas l'exactitude. Par exemple, l'information sonore dégrade les performances (de 89.4% à 88.2%). Une raison peut être le haut taux d'erreurs important du système de classification son/parole (25%). Cependant, l'introduction d'information *a priori* permet toujours d'améliorer les performances.

6.4.4 Discussion sur les résultats de la localisation

Cette section présente le méthode de fusion d'informations multisources multimodales et temporelles par propagation d'activation dans un réseau dynamique qui permet de modéliser la connaissance statique (liée à l'organisation de l'espace) et dynamique (temporelle). Cette approche permet de fusionner des sources non visuelles avec une grande flexibilité et n'impose pas une perception continue de l'espace ni de patron d'événements. De plus, cette méthode permet de prendre en compte facilement des connaissances *a priori* dans le réseau afin d'améliorer les performances. Cette approche a fait l'objet de plusieurs publications (Chahuara et coll., 2010), (Chahuara et coll., 2011a), (Chahuara et coll., 2011c), (Chahuara et coll., 2011b).

Les résultats montrent que la fusion d'information par propagation d'activation multisource présente un intérêt même dans le cas où les sources ont une très bonne précision, comme dans le cas des capteurs infrarouges (mais avec une sensibilité imparfaite). L'utilisation de sources de localisation plus incertaines, telles que la détection de parole, peut alors améliorer les performances. Malgré l'ambiguïté inhérente aux microphones omnidirectionnels, la prise en compte de la connaissance *a priori* du lien entre un son et un microphone permet de corriger un grand nombre d'erreurs de localisation. Cependant, l'utilisation unique des informations de parole conduit cette source à être très peu sensible. Dans notre étude, quasiment 60% des confusions sont dues à des périodes durant lesquelles le sujet ne parle pas, n'utilise pas d'objet et dont les mouvements ne sont pas captés par les PID. Nous prévoyons donc d'inclure la classification de différents types de sons afin de capter les mouvements générateurs de bruits qui ne peuvent être détectés par les PID.

6.5 Évaluation de la méthode de Reconnaissance d'Activité

L'expérimentation 1 dans l'appartement DOMUSndexPlateforme !DOMUS a été menée pour obtenir des données d'activité de vie quotidiennes afin d'évaluer la méthode de reconnaissance d'activités par MLN, presentée dans la section 4.4.2. Le modèle reconnaissance a comme entrée les résultats de localisation. Pour l'algorithme de localisation, l'apprentissage de la structure du réseau s'est fait sur 5 des 21 sujets considérés. Ces sujets ont été retirés du corpus de test pour éviter un biais dans le calcul des performances.

Deux évaluations ont été menées avec le corpus décrit, donnant lieu a deux publications : Chahuara et coll. (2012b) et Chahuara et coll. (2012a).

6.5.1 Évaluation 1

À partir des enregistrements, les données correspondant aux activités énoncées dans le tableau 4.2, sauf l'activité communication, ont été extraites automatiquement, et ce pour différentes tailles de fenêtres temporelles avec différents recouvrements. L'ensemble des classes à classifier est composé des activités plus la classe rejet qui contient principalement les temps entre chacune des activités du scénario.

Les attributs utilisés sont un sous ensemble de ceux montrés dans le tableau 4.2. Ils n'incluent pas l'information de son, parole, et l'agitation. Dans cette première étude nous n'avons pas considéré l'activité Communication. Le détail des attributs choisis sont décrits dans (Chahuara et coll., 2012b).

Dans le corpus on constate un déséquilibre entre le nombre d'individus de chaque classe. Par exemple, la répartition des échantillons pour un fenêtrage d'une minute avec un recouvrement de 25% est respectivement (cf. classes du tableau 4.2) : 210, 160, 27, 232, 42, 88, 151. Ceci s'explique par la différence entre les durées moyennes de chacune des activités (p.ex., manger et se laver les mains).

La stratégie d'apprentissage et de test a été effectuée par *validation-croisée*, divisant le corpus d'apprentissage en 5 parties. Les résultats globaux sont les moyennes des 5 exécutions. Nous avons comparé la performance du MLN avec un classificateur qui a été aussi utilisé pour la reconnaissance d'activités : les séparateurs à vaste marge (cf. section 4.2.1.1) (Fleury et coll., 2010). Les performances de ce classificateur ont été évaluées de la même manière.

Le tableau 6.5 montre le taux de bonnes classifications avec les trois implémentations de MLN avec et sans la classe rejet (dans le cas sans classe rejet, l'apprentissage/test se fait sur six classes). Les résultats du MLN naïf sont meilleurs que les autres méthodes. La classe rejet est difficile à classifier parce qu'elle est extrêmement variable et difficile à modéliser.

En première analyse, il apparaît clairement que les activités à reconnaître dans notre corpus sont souvent liées à un ensemble très réduit d'attributs. Par exemple, la présence dans les toilettes suffit pour expliquer l'activité hygiène. Malgré sa simplicité, le modèle MLN naïf mis en œuvre est adapté pour la reconnaissance d'activités parce qu'il couvre exhaustive-

ment toutes les relations possibles entre capteurs et activités.

Concernant MLN+PLI1, cette méthode n'a pas assez d'exemples positifs et n'offre donc pas de règles utiles pour expliquer une activité. Lorsque le réseau de Markov est instancié, il ne considère pas l'existence des activités qui n'ont pas été apprises. La méthode MLN+PLI2, dont le temps d'apprentissage est beaucoup plus long que MLN+PLI1, offre un ensemble plus conséquent de règles couvrant toutes les activités possibles. Cette augmentation du nombre de règles explique la différence de performance entre ces deux méthodes.

Rejet	Sans						Avec					
Fenêtrage	60 secs			120 secs			60 secs			120 secs		
Recouvrement	0	25%	50%	0	25%	50%	0	25%	50%	0	25%	50%
SVM	64,9	67,3	68,6	64,8	63,8	65	51,5	54,9	59,3	55,4	54,8	56,4
MLN naïf	65,9	70,8	72,8	76,2	65,4	67,2	51,8	55,9	61,2	54,8	57,7	57,5
MLN+PLI1	40,2	59,8	61,3	47,8	40,8	49,2	45,5	40,8	51,1	31,8	30,4	45,6
MLN+PLI2	64,9	66,8	64,2	60,6	59,8	64,2	51,1	53,3	57,9	50,8	52,0	53,7

TABLE 6.5: Taux de bonnes classifications avec SVM et MLN (avec et sans classe rejet)

Même si les résultats de MLN+PLI1 et MLN+PLI2 ne sont pas les meilleurs, ces méthodes permettent d'obtenir un ensemble de règles utiles pour comprendre la relation entre les informations des capteurs et les activités à reconnaître telles que celles obtenues ci-dessous :

$PourcentagePresenceCuisine(90-100\%) \wedge TempsDansLocalisationPresente(8-19min) \wedge VariationTemperatureChambre(0-0.05C^{\circ}) \rightarrow Activite(PreparerRepas)$

$PourcentagePresenceChambre(90-100\%) \wedge VariationCO2Chambre(3-9ppm) \rightarrow Activite(Dormir)$

$PourcentagePresenceBureau(90-100\%) \rightarrow Activite(Detente)$

6.5.2 Évaluation 2

Dans cet étude (Chahuara et coll., 2012a), nous avons inclu l'information de son, de parole, et d'agitation comme information inférée. Nous avons ajouté la communication comme activité à classifier. Les attributs utilisés et les classes correspondent à ceux décrits dans le tableau 4.2.

Le stratégie d'apprentissage et de test est une *Validation croisée* comme dans l'évaluation précédente. Pourtant, dans cette étude nous avons organisé les partitions d'une autre façon : à chaque itération les donnés d'un sujet sont choisies pour le test et le reste du corpus est utilisé pour l'apprentissage. L'objectif de cette stratégie est d'analyser à quel point les données d'apprentissage d'un groupe de sujets peuvent être généralisées pour être appliquées à la reconnaissance de l'activité d'un sujet différent.

Nous avons utilisé une seule taille de fenêtre et de recouvrement. La taille de la fenêtre temporelle a été fixée de façon empirique à une minute. De plus, étant donné que quelques activités peuvent dépasser la taille d'une fenêtre, un chevauchement de 25% a été considéré.

Étant donnée que dans l'évaluation précédente la méthode MLN qui a donné les meilleurs

résultats est **MLN naïf**, dans ce cas nous avons appliqué seulement cette version. En plus des SVM, nous avons utilisé les réseaux bayésiens naïfs comme méthode de comparaison. Cette méthode est une simplification des réseaux bayésiens qui se base sur la même hypothèse que notre approche : l'indépendance probabiliste des attributs étant donnée la classe cible. D'ailleurs, cette méthode a été appliquée pour la reconnaissance d'activité avec des résultats très acceptables Portet et coll. (2009).

Les résultats que nous avons trouvés sont résumés dans le tableau 6.6. L'exactitude générale obtenue avec le MLN est largement supérieur à celle des autres modèles. La classe rejet qui correspond aux fenêtres temporelles où il n'y a pas d'activités spécifiques, donne le résultat le plus bas. Il est possible de vérifier, par rapport à l'évaluation 1, que l'inclusion des informations sonores et d'agitation améliorent la reconnaissance des activités.

TABLE 6.6: Exactitude général, précision et rappel pour chaque classe

Méthode	SVM		Naive Bayes		MLN	
Exactitude	59.64		66.1		85.25	
	Précision	Rappel	Précision	Rappel	Précision	Rappel
Preparer Repas	64.8	71.0	75.1	79.8	90.4	91.9
Rangement	40.0	39.0	58.3	56.4	75.1	86.9
Hygiène	55.8	57.4	67.4	61.7	82.6	80.9
Communication	83.7	71.9	40.9	47.4	100.0	82.5
S'habiller	32.3	41.1	13.3	11.8	85.3	56.9
Dormir	57.6	60.1	60.1	74.3	84.7	82.2
Se détendre	81.5	73.5	82.4	70.8	90.2	92.2
Rejet	10,2	8.2	19.7	17.9	63.6	25.0

Nous considérons que malgré sa simplicité le modèle MLN naïf est approprié pour la classification des activités. D'après les résultats, il semble pertinent de supposer que l'évidence de la réalisation d'une activité est indépendante de l'information des autres capteurs sachant l'activité en cours. C'est ainsi que notre modèle donne des bons résultats à partir des règles qui modélisent indépendamment l'influence de chaque valeur de capteur pour reconnaître une activité. Nous montrons ci-dessous quelques règles du MLN ayant les poids plus grands :

2.17 : $PourcentagePresenceBedroom(90-100\%) => class(Dormir)$

2.14 : $PourcentageAgitationElectricite(80-100\%) => class(Rangement)$

1.94 : $ParoleBureau(8-13) => class(Communication)$

1.85 : $SounBedroom(8-12) => class(Habillage)$

Ces règles peuvent nous aider à comprendre les connaissances induites par le processus d'apprentissage. Le poids des règles est un bon indicateur de l'importance de chaque règle pour reconnaître une activité. Comme nous l'avons souligné précédemment, la localisation de l'habitant est un attribut essentiel, dû au fait que la plupart des activités sont réalisées dans une pièce spécifique. D'autres informations, peuvent complémenter l'information de la localisation, comme dans la deuxième règle l'utilisation intensive d'électricité peut mettre en évidence l'utilisation de l'aspirateur pour déduire l'activité de Rangement. Également, la reconnaissance de la parole dans la pièce où se trouve le téléphone est une forte évidence de

l'activité Communication.

6.5.3 Discussion sur la reconnaissance d'activité

Dans la première évaluation, les résultats entre les MLN et les SVM sont relativement bas et proches (autour des 65% sans classe rejet et 55% avec classe rejet), cependant une légère amélioration est à noter dans le cas des MLN naïfs. Cette deuxième méthode permet en plus de mieux comprendre la manière dont la séparation est faite entre les classes. Pour les MLN avec apprentissage fondé sur la PLI, le nombre d'exemples de certaines classes et le déséquilibre entre classes peut également être responsable de ces mauvais résultats.

Les méthodes MLN+PLI1 et MLN+PLI2 permettent d'obtenir un ensemble de règles vérifiables qui pourraient être réutilisées pour l'apprentissage dans d'autres habitats de façon à constituer une base de règles *a priori* génériques voir même être transférée directement dans un classificateur. Le développement de méthodes MLN+PLI semble être un axe prometteur pour acquérir des classificateurs réutilisables dans d'autres environnements que ceux dans lesquels ils ont été appris.

Les résultats de la deuxième évaluation où nous avons inclue les informations de son et parole sont meilleures, ce qui prouve l'importance de l'information sonore pour la reconnaissance d'activité.

Ces résultats sont en partie inférieurs à d'autres résultats de la littérature, cependant ce fait peut s'expliquer par les capteurs utilisés, qui sont plus facilement installables (car tous sur bus KNX) mais aussi beaucoup moins informatifs que certains capteurs portés (contraignants). Cependant, les données acquises dans le cadre de cette étude montrent un plus fort réalisme des conditions d'enregistrement (ambiguïté, activités interrompues, activité impliquant plusieurs pièces, etc.) que celui de la plupart des autres travaux de la littérature qui soit ne considèrent pas le problème de détection soit ne considèrent pas de classe rejet.

En résumé, les principaux avantages de l'approche MLN sont : la capacité de représenter en détail les relations entre les attributs et les activités, la lisibilité du modèle permettant de comprendre quels capteurs expliquent l'occurrence d'une activité, et la possibilité d'inclure des connaissances complémentaires, sous forme de règles logiques, comme il a été fait avec les relations temporelles entre fenêtres.

6.6 Évaluation de la reconnaissance de situations et commandes vocales

L'intérêt de la reconnaissance de situations dans nos recherches est plutôt la définition des règles *événement-condition-action (ECA)* qui soient faciles à inclure dans la base de connaissances et qui servent à exprimer le comportement réactive du contrôleur. Dans ce cas, nous ne traitons pas l'incertitude des informations. Deux situations à reconnaître ont été mises en place dans le scénario de l'expérimentation 2. La description de ces situations avec leurs représentations par SWRL (cf. 4.5.2) sont montrés au dessous :

1. **Situation 1**. Événement : La personne ouvre la porte principale de l'appartement pour sortir. Condition : les fenêtres de la chambre sont ouvertes. Action : Envoyer un message par le synthétiseur vocal.

   Représentation par SWRL

   DeviceEvent(?d), has_associated_object(?d, door),
   takes_place_in(?d, kitchen),state_value(?d, open),
   Window(?w), located_in(?w, bedroom),Application(?a),
   has_application(?w, ?a),curret_state(?a,on)
   → current_state(BedroomWindowsOpen, detected)

2. **Situation 2**. Événement : Les lumières de la chambre se éteints. Condition : les rideaux et les fenêtres sont fermés, et la porte principale est ouverte. Action : Envoyer un message par le synthétiseur vocale.

   Représentation par SWRL

   DeviceEvent(?l), has_associated_object(?l, light),
   takes_place_in(?l, bedroom),state_value(?d, off),
   Window(?w), located_in(?w, bedroom),Application(?a1),
   has_application(?w, ?a1),curret_state(?a1,off)
   Blind(?b), located_in(?b, bedroom),Application(?a2),
   has_application(?b, ?a2),curret_state(?a2,off)
   Door(?d), located_in(?d, kitchen),Application(?a3),
   has_application(?bd, ?a3),curret_state(?a3,on)
   → current_state(MainDoorOpen, detected)

Le scénario contient la réalisation des situations une fois chacune. Pourtant, des fois les sujets n'ont pas suivi le scénario de façon tel que les conditions pour déclencher les situations soient satisfaites. Lors des expérimentations avec les 13 sujets, ces situations ont été déclenchées 8 et 5 fois respectivement. Dans tous les cas, la réaction du contrôleur intelligent a été immédiat une fois que l'événement a été reconnu et la condition validée. Le test dans des conditions réelles de ces situations sert à prouver qu'elles représentent un cadre idéal pour démarrer le processus de prise de décision. Ainsi, à la place d'exécuter une action directement quand une situation est reconnue, le contrôleur pourrait analyser les informations contextuelles et se servir d'un modèle de décision pour choisir l'action la plus pertinente.

6.7 Évaluation de la méthode de prise de décision

Pour évaluer notre modèle de prise de décision fondé sur les MLN, nous avons utilisé les informations obtenues de l'expérimentation 2. Dans les 13 scénarios déroulés dans cette expérimentation, 4 situations de décision ont été mis en place. Toutes les situations ont été déclenchées par l'ordre vocal « Nestor/Maison, Allume la lumière ». Dans toutes les situations il y a dans la pièce 2 lumières, l'un ayant une intensité plus forte que l'autre.

1. **Situation de décision 1**. La personne prend son repas assise à la table de la cuisine. La lumière la plus appropriée et celle de la lampe sur la table.

2. **Situation de décision 2**. La personne est en train de ranger dans la chambre. La lumière la plus appropriée et celle du plafond.

3. **Situation de décision 3**. La personne est en train de nettoyer l'évier et faire la vaisselle dans la cuisine. La lumière la plus appropriée et celle du plafond.

4. **Situation de décision 4**. La personne vient de finir la sieste dans la chambre. La lumière la plus appropriée est celle des lampes de chevets.

6.7.1 Modèle pour la prise de décision

Nous montrons au dessous le modèle MLN développé pour trouver la lumière avec la localisation et l'intensité pertinente selon chaque situation. Ce modèle se base sur l'exemple de la section 5.4.

Variables	**Domaine**
degree	{bas,haut }
con	{bas,moyen,haut }
loc	{chambre,cuisine,salleDeBain,bureau }
act	{dormir,ranger,repas, hygiene,habillage, communication, detente }

Type	**Prédicats**
Action	Intensite, LocalisationLumiere
État	Agitation, Proche, Localisation,Activite
Utilité	EndroitIdeal, Confort

Poids	**règles**
3,35	$LocalisationLumiere(loc) \wedge Localisation(loc)$ $\rightarrow EndroitIdeal(bon)$
0,12	$LocalisationLumiere(loc1) \wedge Localisation(loc2) \wedge Proche(loc1, loc2)$ $\rightarrow EndroitIdeal(moyen)$
2,44	$LocalisationLumiere(loc1) \wedge Localisation(loc2) \wedge loc1 \neq loc2$ $\rightarrow EndroitIdeal(bas)$
1,46	$Activite(act) \wedge Agitation(act, degree) \wedge Intensite(degree)$ $\rightarrow Confort(haut)$
-0,79	$Activite(act) \wedge Agitation(act, degree1) \wedge Intensite(degree2) \wedge degree1 \neq degree2$ $\rightarrow Confort(moyen)$
-0,09	$Activite(act) \wedge Agitation(act, degree1) \wedge Intensite(degree2) \wedge degree1 \neq degree2$ $\rightarrow Confort(bas)$

Valeurs d'utilité

U(EndroitIdeal(mauvais))=-2	U(EndroitIdeal(acceptable))=0	U(EndroitIdeal(bon))=2
U(Confort(bas))=-3	U(Confort(moyen))=0	U(Confort(haut))=3

Données factuelles

Proche(cuisine,chambre)	Proche(chambre,cuisine)	Proche(chambre,salleDeBain)
Proche(salleDeBain,chambre)	Proche(bureau,chambre)	Proche(chambre,bureau)
Agitation(dormir,bas)	Agitation(repas,bas)	Agitation(ranger,haut)
Agitation(hygiene,haut)	Agitation(habillage,haut)	Agitation(communication,haut)
Agitation(detente,haut)		

Chaque situation de prise de décision apparaît une seule fois dans le scénario. Nous comptons uniquement sur les données des 13 sujets, ce qui résulte insuffisante pour trouver le poids optimal des règles. En fait, l'obtention d'un corpus exclusivement des situations de prise de décision serait trop chère en temps d'expérimentation et ressources. Donc, la solution que nous avons trouvée est la création d'un corpus d'apprentissage artificiel de 200 instances, la plupart desquelles expriment les meilleures localisations et intensités, comme il a été décrit pour les 4 situations de décision. Cependant, nous avons aussi inclus dans ce corpus des configurations qui peuvent être contradictoires. Par exemple, si dans la plupart des instances de la situation (1) la meilleure lumière pour manger est celle au dessus de la table, dans quelques instances celle du plafond est acceptable et très rarement la lumière sur la table est considérée mauvaise pour le confort. Les poids que nous avons appris, sont ceux que nous montrons au dessus dans le modèle. À partir des poids, il est possible comprendre le contenu du corpus : la bonne localisation et intensité sont toujours préférées. Une localisation incorrecte de la lumière ne serait jamais convenable ; par contre, une intensité incorrecte n'entraîne pas un grand risque pour le confort.

Les valeurs des utilités gardent une proportion par rapport à ces équivalences qualitatives (mauvais=-2, acceptable=0, bon=2). Pourtant, pour le confort, qui est plutôt une conséquence de l'intensité de la lumière, nous avons décidé de donner une utilité plus grande car elle dépend de l'activité en cours de réalisation, une variable plus incertaine et difficile à déterminer que la localisation.

6.7.2 Determination des informations contextuelles

Nous avons appliqué le modèle de localisation décrit dans la section 4.3. La localisation de la personne, à l'instant qu'elle prononce un ordre vocale, peut être déduite avec une certitude très proche à 100% par la fusion des informations des microphones ayant le meilleur RSB et les capteurs de présence. Alors, au moment de recevoir l'ordre, le contrôleur ne considère pas la localisation comme une variable incertaine pour la prise de décision. La localisation optimale de la lumière est toujours celle donnée par l'algorithme de réseau dynamique qui est, à cet instant, exacte. Bien que la localisation dans ce cas, où la prise décision est déclenchée par une commande vocale, est facile à déterminer ; dans d'autres situations où les événements donnent une évidence moins fortes pour le réseau dynamique, la localisation serait plus affectée par l'incertitude.

L'activité en cours est, par contre, une variable sujet à l'incertitude. La version naïve de MLN, exposée dans la section 4.4.2, a été utilisée. Les poids du MLN pour la reconnaissance d'activité ont été appris avec le corpus de l'expérimentation 1. Nous nous intéressons uniquement à l'inférence de l'activité à l'instant où les commandes vocales sont prononcées. À ce fin, il est nécessaire d'analyser, pour les données obtenues de chacun des 13 sujets, les fenêtres temporelles (fixées à 60 secondes) qui précédent les 4 situations de décision. Comme il est indiqué dans les dernières trois règles du modèle de décision, l'inférence correcte de l'activité est importante pour fournir une intensité de lumière appropriée. Les 52 instances pour la reconnaissance d'activité sont repartis dans 3 activités : dormir (13 instances), repas (13 instances), ranger (26) ; pourtant, le modèle MLN considère les sept activités que nous avons définis dans tout notre travail de recherche. Le tableau 6.8 montre la matrice de confusion pour la reconnaissance d'activité.

Activité/Estimée MLN	Repas	Ranger	Habillage	Dormir	Détente	Hygiène	Communication
Repas	7	6	0	0	0	0	0
Ranger	3	17	1	1	4	0	0
Dormir	0	2	1	9	1	0	0

TABLE 6.7: Matrice de confusion pour la reconnaissance d'activité lors de la prise de décision

La confusion la plus importante se situe entre l'activité **Ranger** et **Repas**. Ainsi, presque la moitié de des activités **Repas** ont été reconnues comme **Ranger**. Il faut remarquer que ces deux activités, quand elles sont effectués dans la cuisine, partagent plusieurs caractéristiques comme le bruit l'agitation sonore produite par la vaisselle. En plus, dans l'expérimentation 2 le nettoyage n'utilise pas l'aspirateur, dont le bruit rendait plus facile sa reconnaissance dans l'expérimentation 1 (cf. 6.5). L'exactitude finale de la reconnaissance est de 63,46%. Nous considérons ce taux d'acceptation est bon étant donnée que l'apprentissage du modèle a été fait à partir d'expérimentations où les activités n'ont pas été faites exactement de la même façon.

6.7.3 Résultats avec le modèle de décision

Le tableau 6.8 montre le pourcentage du nombre d'instances où le contrôleur a pris la décision correcte pour chaque situation définie dans l'expérimentation 2. Les résultats avec et sans la prise en compte de l'incertitude des variables contextuelles sont présentés. Dans le deuxième cas, l'activité et la localisation avec les meilleures probabilités sont prises et elles sont considérées comme des évidences factuelles. En fait, comme remarqué dans la section 6.7.2, l'exactitude de la localisation est dans tous les cas proche de 100%, donc l'incertitude est due surtout à la connaissance de l'activité.

Situation/Estimation de l'utilité	Avec l'incertitude de l'activité	Sans l'incertitude de l'activité
Situation 1	46%	54%
Situation 2	100%	100%
Situation 3	84%	69%
Situation 4	54%	62%
Total	71%	71%

TABLE 6.8: Pourcentage d'instances de décision avec l'action correcte

La performance la plus base est celle de la prise de décision pour la situation 1. Cela s'explique par la confusion dans la reconnaissance de l'activité **repas** qui est confondue souvent avec **ranger**. L'activité **ranger**, par contre, a été bien reconnue, surtout dans la chambre, et par conséquence, le contrôleur arrive à prendre la meilleure décision. Les résultats avec et sans évidence incertaine sont assez similaires parce que les instances où la décision diffère selon ces méthodes sont seulement 4 parmi les 52. Un exemple d'une instance où l'évidence incertaine a été utile pour trouver la meilleure solution est la situation 3 pour le sujet 12. Dans cette instance les activités avec les probabilités plus fortes sont : hygiène (0.20), habillage (0.16), dormir (0.28), et détente (0.17) ; quand la vrai activité est ranger (0.08) dans la cuisine. Il faut noter que si la localisation est fiable lors de la réception de la commande vocale grâce à l'analyse des informations sonores de parole, la reconnaissance d'activité nécessite l'analyse d'une fenêtre temporelle et dans cette intervalle il peux exister des erreurs de localisation ; ce qui explique le fait de reconnaître l'activité **dormir** quand la personne est en train de manger. Dans l'instance de cet exemple, il existe une grande incertitude par rapport a l'activité en cours, par contre le poids de probabilité des activités ayant une agitation haute aide au contrôleur de préférer une intensité de lumière haute, malgré que l'activité la plus probable est **dormir**.

6.7.4 Discussion sur les résultats de prise de décision

Dans cette section, nous avons présenté un modèle pour la prise de décision développé avec les réseaux logiques de Markov. Malgré sa simplicité, il s'agit d'un modèle assez expressif car les relations entre les éléments autour du processus de décision sont faciles à

comprendre et il possible d'exprimer les autres relations à travers les règles logiques avant d'évalue leur pertinence selon leur poids. Le calcul des poids est, par contre, un des points faibles parce qu'il demande l'obtention d'un corpus d'apprentissage de taille significative, comme dans la plupart des méthodes probabilistes.

Des résultats exposés, nous pouvons conclure que la décision correcte dépend fortement de la certitude des informations contextuelles. Dans les situations sur lesquelles nous avons basé nos études, le fait d'exécuter une action incorrecte n'entraîne pas des conséquences graves pour l'habitant. En cas, par exemple, de la détection d'une situation de détresse, le risque d'envoyer une fausse alarme peut être très grand selon l'incertitude des informations contextuelles, et cela peut avoir un coût énorme si la sécurité de l'habitant est en jeu. Le cadre de travail que nous avons développé permet d'exprimer ces risques par l'inclusion des valeurs d'utilité négatives en cas d'une mauvaise décision ; pourtant, l'action finale est toujours influencée par le contexte. D'ailleurs, bien que l'inclusion de l'incertitude des informations contextuelles, dans nos recherches, n'ait pas entraîné une amélioration dans les résultats, son utilisation dans la prise de décision est quand même importante. Ainsi, nous pourrions spécifier que s'il existe une grande incertitude lors d'une situation de détresse, une confirmation doit être demandée à l'habitant.

CHAPITRE 7

Conclusion et perspectives

Dans ce mémoire de thèse, nous avons présenté une nouvelle approche pour la réalisation d'un contrôleur intelligent de la domotique. L'architecture fonctionnelle ainsi que les méthodes de représentation de la connaissance du domaine ont été présentés en détail. Une part importante du doctorat à consister à définir les différents niveaux d'abstraction qui permettent d'inférer les informations du contexte nécessaire à une prise de décision correcte. Pour chaque type d'information à extraire, une méthode état de l'art a été adaptée et validée par des expérimentations. Ces expérimentations ont fait partie intégrante du travail de thèse. D'une manière générale, cette thèse apporte une contribution au domaine de l'intelligence ambiante et des habitats intelligents.

La définition complète de l'ontologie et la division du modèle de représentation en deux couches sémantiques est une approche particulièrement utile pour adapter le système à de nouveaux environnements intelligents. Les connaissances ne se limitent pas à la description des éléments physiques dans la maison intelligente ; mais elles contiennent aussi des éléments abstraits et les relations existantes entre les différents concepts du domaine. L'extension du modèle de connaissance par l'inclusion de règles logiques a montré toute sa pertinence pour la représentation des situations à risque ou de détresse.

Le travail de thèse a également consisté à définir des algorithmes d'inférence pour fournir de l'information contextuelle à l'étage de décision. Le contexte permet de désambiguïser les situations à partir desquelles le contrôleur doit réagir à une commande domotique ou à une situation à risque. La localisation et l'activité sont les informations contextuelles les plus importantes dans une maison intelligente en parfait accord avec la plupart des travaux cités dans l'état de l'art. La fusion d'informations par réseaux de propagation et la classification par réseaux logiques de Markov ont été étudiées pour inférer ces informations à partir des données brutes du réseau domotique.

Pour prendre en compte l'incertitude d'une situation lors d'une prise de décision, un modèle à base de Réseau Logique de Markov a été conçu. Étant donné un modèle graphique, les éléments du processus de décision, tels que l'utilité et les actions possibles, sont faciles à reconnaître et à modifier. Finalement, le cadre de travail, fondé sur des formules logiques, permet de représenter des relations complexes entre les variables du domaine de décision.

Concernant la conception et réalisation d'un contrôleur intelligent pour piloter un environnement perceptif à partir des informations obtenues du réseau domotique, la thèse a proposée une architecture hiérarchique des différents composants du contrôleur de façon à

ce que la sortie d'un module d'inférence informe un autre module traitant les informations de plus haut niveau. Cela n'en fait pas pour autant une simple architecture « pipeline » car le système permet l'intégration de n'importe quel module d'inférence ou autres méthodes de raisonnement qui pourra communiquer avec les autres via l'ontologie.

Enfin, un travail important d'expérimentations a été entrepris pour concevoir les protocoles en accord avec les objectifs de la thèse, préparer le matériel d'expérimentation (assez lourd à mettre en place dans un habitat intelligent), préparer les documents à destination des participants, préparer les scénarios d'expérimentation, concevoir les outils d'expérimentation, accueillir et mener les enregistrements, annoter les heures de données recueillies et préparer celles-ci pour servir de base d'apprentissage et de tests.

Comme il a été exposé dans l'état de l'art de cette thèse, les systèmes d'intelligence ambiante présentés dans la littérature sont souvent fondés sur une de deux approches principale de l'intelligence artificielle : la logique, ayant l'avantage d'exprimer des relations complexes entre les entités du domaine dans un cadre formel et avec une grande expressivité ; et l'approche probabiliste, qui est moins expressive mais qui permet de traiter l'incertitude des informations et qui est, en conséquence, mieux adaptée à des applications réelles telles que les environnements intelligents. Nous avons basé notre travail sur une approche originale qui prend non seulement les caractéristiques de la logique mais aussi celles de l'approche probabiliste : les modèles statistico-relationnels. Dans cette approche, il est possible de mettre en place un ensemble des règles logiques qui décrivent les connaissances et de leur assigner des poids pour exprimer la certitude que l'on a sur ces règles. Ainsi, il est possible de modéliser non seulement l'incertitude des variables du problème mais aussi celle de la structure du modèle, c.-à.d. des relations existantes dans le modèle.

Étant donné l'importance d'obtenir des modèles de connaissances lisibles et traitables par les systèmes informatiques, nous considérons que les modèles statistico-relationnels et leur techniques d'apprentissage automatique sont très prometteurs pour le développement de l'intelligence ambiante et la réalisation de systèmes qui puissent être mis en œuvre dans des situations réelles.

En se basant sur cette approche, il est possible d'identifier plusieurs pistes à explorer.

Les environnements perceptifs offrent un domaine de connaissances très riche en concepts et en relations qui devraient être exploités pour améliorer les inférences et la prise de décision. Nous avons remarqué dans nos travaux de reconnaissance d'activité, que l'assignation de poids aux règles d'un modèle statistique-relationnel, sert non-seulement à construire un modèle de classification, mais également à donner un critère d'évaluation des règles très utile pour analyser les connaissances exprimées dans le modèle et mieux comprendre celui-ci. Cependant, la structure de ces modèles a souvent été défini *a priori*, sauf dans le cas de l'application d'ILP pour acquérir automatiquement les règles du MLN. Nous considérons qu'il est intéressant d'appliquer d'autres méthodes d'apprentissage relationnel pour extraire des relations importantes stockées dans les corpus d'apprentissage, afin d'améliorer la précision de nos modèles et gagner une meilleure compréhension de l'environnement perceptif.

Les méthodes de prise de décision et de reconnaissance d'activité présentées dans cette thèse se basent sur une représentation logique elles n'utilisent pas directement les relations sémantiques exprimées dans les ontologies. Si les règles faisant partie de ces modèles étaient partagées dans une même base de connaissance, les modules d'inférences pourraient réutiliser les connaissances déjà apprises et les connaissances *a priori* pourraient également être partagées. Nous pensons qu'il serait intéressant de développer des méthodes pour intégrer la structure des modèles statiques relationnels, ainsi que les poids de confiances de leurs règles, dans les ontologies, dans le but d'atteindre une cohérence dans les connaissances représentées dans ces modèles.

Nous avons utilisé l'information sonore uniquement pour la localisation de l'habitant à travers le RSB, et ensuite pour la reconnaissance des ordres vocaux. Cependant, l'analyse et la classification des informations sonores peuvent fournir des informations très utiles à mettre en place dans les tâches de raisonnement. Par exemple la reconnaissance de sons de vaisselle pourrait diminuer largement l'ambiguïté lors de la reconnaissance de l'activité de l'habitant ou de sa localisation.

Bibliographie

ABOWD, G. D., DEY, A. K., BROWN, P. J., DAVIES, N., SMITH, M. et STEGGLES, P. (1999). Towards a better understanding of context and context-awareness. Dans *Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, HUC '99, pages 304–307, London, UK, UK. Springer-Verlag.

AGGARWAL, J. et RYOO, M. (2011). Human activity analysis : A review. *ACM Comput. Surv.*, 43:1–43.

ALDRICH, F. (2003). Smart homes : Past, present and future. Dans HARPER, R., éditeur : *Inside the Smart Home*, pages 17–39. Springer London.

ALEXANDER, G., RANTZ, M., SKUBIC, M., KOOPMAN, R., PHILLIPS, L., GUEVARA, R. et MILLER, S. (2011). Evolution of an early illness warning system to monitor frail elders in independent living. *J Healthc Eng*, 2(3):337–363.

ALLEN, J. F. et FERGUSON, G. (1994). Actions and Events in Interval Temporal Logic. *Journal of Logic and Computation*, 4(5):531–579.

ALOISE, F., ARICÒ, P., SCHETTINI, F., RICCIO, A., RISETTI, M., SALINARI, S., MATTIA, D., BABILONI, F. et CINCOTTI, F. (2011). A new p300 no eye-gaze based interface : Geospell. Dans BABILONI, F., FRED, A. L. N., FILIPE, J. et GAMBOA, H., éditeurs : *BIOSIGNALS*, pages 227–232. SciTePress.

ANAGNOSTOPOULOS, C., NTARLADIMAS, Y. et HADJIEFTHYMIADES, S. (2007). Situational computing : An innovative architecture with imprecise reasoning. *Journal of Systems and Software*, 80(12):1993–2014.

ARTIKIS, A., SKARLATIDIS, A., PORTET, F. et G., P. (to appear). Logic-based event recognition. *Knowledge Engineering Review*.

ASWATH, D., D'CUNHA, J., AHMED, S. T. et DAVULCU, H. (2005). Boosting item keyword search with spreading activation. Dans *IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*, pages 704–707.

AUGUSTO, J. C. (2009). Ambient intelligence : Opportunities and consequences of its use in smart classrooms. *Italics*, 8(2).

AUGUSTO, J. C., NAKASHIMA, H. et AGHAJAN, H. (2010). Ambient intelligence and smart environments : A state of the art. Dans *Handbook of Ambient Intelligence and Smart Environments*, pages 3–31.

AUGUSTO, J. C. et NUGENT, C. D. (2004). The use of temporal reasoning and management of complex events in smart homes. *European Conference on Artificial Intelligence*, pages 778–782.

AUGUSTO, J. C. et NUGENT, C. D. (2006). Smart homes can be smarter. Dans *Designing Smart Homes*, pages 1–15.

BAADER, F., HORROCKS, I. et SATTLER, U. (2008). Description Logics. Dans VAN HARMELEN, F., LIFSCHITZ, V. et PORTER, B., éditeurs : *Handbook of Knowledge Representation*, chapitre 3, pages 135–180. Elsevier.

BADII, A. et BOUDY, J. (2009). CompanionAble - integrated cognitive assistive & domotic companion robotic systems for ability & security. Dans *1st Congres of the Société Française des Technologies pour l'Autonomie et de Gérontechnologie (SFTAG'09)*, pages 18–20, Troyes.

BAHADORI, S., CESTA, A., GRISETTI, G., IOCCHI, L., LEONE, R., NARDI, D., ODDI, A., PECORA, F. et RASCONI, R. (2004). RoboCare : Pervasive Intelligence for the Domestic Care of the Elderly. *Intelligenza Artificiale*, 1(1):16–21.

BENAYOUN, R., MONTGOLFIER, J., TERGNY, J. et LARICHEV, O. (1971). Linear programming with multiple objective functions : Step method(stem). *Mathematical Programming*, 1(2): 366–373.

BERENGUER, M., GIORDANI, M., GIRAUD-BY, F. et NOURY, N. (2008). Automatic detection of activities of daily living from detecting and classifying electrical events on the residential power line. Dans *HealthCom'08, 10th IEEE Int. Conf. on e-Health Networking, Applications & Service*.

BESAG, J. (1975). Statistical Analysis of Non-Lattice Data. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 24(3):179–195.

BIAN, X., ABOWD, G. D. et REHG, J. M. (2005). Using sound source localization in a home environment. Dans *Proceedings of the Third international conference on Pervasive Computing*, pages 19–36, Berlin, Heidelberg. Springer-Verlag.

BILINSKI, P. et BRÉMOND, F. (2012). Contextual statistics of space-time ordered features for human action recognition. Dans *AVSS*, pages 228–233. IEEE Computer Society.

BONARINI, A. et MANIEZZO, V. (1991). Integrating expert systems and decision-support systems : principles and practice. *Knowl.-Based Syst.*, 4(3):172–176.

BOSER, B. E., GUYON, I. M. et VAPNIK, V. N. (1992). A training algorithm for optimal margin classifiers. Dans *Fifth Annual Workshop on Computational Learning Theory, ACM*, pages 144 – 152, Pittsburgh.

BRDICZKA, O., LANGET, M., MAISONNASSE, J. et CROWLEY, J. (2009). Detecting human behavior models from multimodal observation in a smart home. *IEEE Transactions on Automation Science and Engineering*, 6(4):588 –597.

BROWN, P., BOVEY, J. et CHEN, X. (1997). Context-aware applications : from the laboratory to the marketplace. *Personal Communications, IEEE*, 4(5):58–64.

BUCHANAN, B. G. et SHORTLIFFE, E. H., éditeurs (1985). *Rule-Based Expert Systems : The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Reading, MA.

CAROLIS, B. D. et COZZOLONGO, G. (2004). C@sa : Intelligent home control and simulation. Dans *International Conference on Computational Intelligence*, pages 462–465.

CASELLA, G. et GEORGE, E. I. (1992). Explaining the Gibbs Sampler. *The American Statistician*, 46(3):167–174.

CATARCI, T., CINCOTTI, F., LEONI, M. d., MECELLA, M. et SANTUCCI, G. (2009). Smart homes for all : Collaborating services in a for-all architecture for domotics. Dans BERTINO, E. et JOSHI, J. B. D., éditeurs : *Collaborative Computing : Networking, Applications and Worksharing*, numéro 10 de Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pages 56–69. Springer Berlin Heidelberg.

CHAHUARA, P., FLEURY, A., PORTET, F. et VACHER, M. (2012a). Using Markov Logic Network for On-line Activity Recognition from Non-Visual Home Automation Sensors. Dans *Ambient Intelligence*, volume 7683 de *Lecture Notes in Computer Science*, pages 177–192, Pisa, Italy. Springer (Heidelberg).

CHAHUARA, P., FLEURY, A., VACHER, M. et PORTET, F. (2012b). Méthodes SVM et MLN pour la reconnaissance automatique d'activités humaines dans les habitats perceptifs : tests et perspectives. Dans *Actes de la conférence RFIA 2012*, pages 340–347, Lyon, France.

CHAHUARA, P., PORTET, F. et VACHER, M. (2011a). Fusion of Audio and Temporal Multimodal Data by Spreading Activation for Dweller Localisation in a Smart Home. Dans *STAMI Series, Space, Time and Ambient Intelligence*, Barcelona, Spain.

CHAHUARA, P., PORTET, F. et VACHER, M. (2011b). Location of an Inhabitant for Domotic Assistance Through Fusion of Audio and Non-Visual Data. Dans *Pervasive Health*, page 4p., Dublin, Ireland.

CHAHUARA, P., VACHER, M. et PORTET, F. (2010). Localisation d'habitant dans un environnement perceptif non visuel par propagation d'activations multisource. Dans *MajecSTIC 2010*.

CHAHUARA, P., VACHER, M. et PORTET, F. (2011c). Localisation d'habitant dans un espace perceptif par réseau dynamique. Dans *Représentation et Raisonnement sur le Temps et l'Espace (RTE 2011)*, Chambéry, France.

CHAN, H. et DARWICHE, A. (2005). On the revision of probabilistic beliefs using uncertain evidence. *Artificial Intelligence*, 163(1):67–90.

CHAN, M., ESTÈVE, D., ESCRIBA, C. et CAMPO, E. (2008). A review of smart homes- present state and future challenges. *Computer Methods and Programs in Biomedicine*, 91(1):55–81.

CHARLON, Y., BOURENNANE, W. et CAMPO, E. (2011). Mise en oeuvre d'une plateforme de suivi de l'actimétrie associée à un système d'identification. Dans *Symposium Mobilité et Santé (SMS 2011)*.

CHEN, C. et COOK, D. J. (2012). Behavior-based home energy prediction. Dans *Intelligent Environments*, pages 57–63. IEEE.

CHEN, G. et KOTZ, D. (2000). A survey of context-aware mobile computing research. Rapport technique, Hanover, NH, USA.

CHEN, H., FININ, T., JOSHI, A., KAGAL, L., PERICH, F. et CHAKRABORTY, D. (2004). Intelligent agents meet the semantic web in smart spaces. *IEEE Internet Computing*, 8(6):69 – 79.

CHEN, J., KAM, A. H., ZHANG, J., LIU, N. et SHUE, L. (2005). Bathroom activity monitoring based on sound. Dans *Proceedings of Pervasive 2005, International conference*, pages 47–61, Munich, Germany.

CHEN, L. et NUGENT, C. (2009). Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems*, 5(4):410–430.

CHEN, L., NUGENT, C., MULVENNA, M., FINLAY, D., HONG, X. et POLAND, M. (2008). A logical framework for behaviour reasoning and assistance in a smart home. *International Journal of Assistive Robotics and Mechatronics*, 9(4).

CHEN, P. P. (1976). The Entity-Relationship Model : Toward a Unified View of Data. *ACM Transactions on Database Systems*, 1:9–36.

CHUA, S.-L., MARSLAND, S. et GUESGEN, H. (2009). Behaviour Recognition from Sensory Streams in Smart Environments. Dans NICHOLSON, A. et LI, X., éditeurs : *AI 2009 : Advances in Artificial Intelligence*, volume 5866, chapitre 67, pages 666–675. Springer Berlin Heidelberg, Berlin, Heidelberg.

COOK, D. J. et DAS, S. K. (2005). *Smart environments - technology, protocols and applications*. Wiley.

COOK, D. J. et DAS, S. K. (2007). How smart are our environments ? an updated look at the state of the art. *Pervasive and Mobile Computing*, 3(2):53–73.

CORNALBA, C., BELLAZZI, R. et BELLAZZI, R. (2008). Building a normative decision support system for clinical and operational risk management in hemodialysis. *IEEE Transactions on Information Technology in Biomedicine*, 12(5):678 –686.

CORTES, C. et VAPNIK, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

CRESTANI, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6):453–482.

DALAL, S., ALWAN, M., SEIFRAFI, R., KELL, S. et BROWN, D. (2005). A rule-based approach to the analysis of elders activity data : Detection of health and possible emergency conditions. Dans *AAAI Fall 2005 Symposium*.

DEAN, M. et SCHREIBER, G. (2004). OWL web ontology language reference. W3C recommendation, W3C.

DEY, A. K. (1998). Context-aware computing : The cyberdesk project. Dans *AAAI 1998 Spring Symposium on Intelligent Environments*, pages 51–54, Palo Alto. AAAI Press.

DEY, A. K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, 5(1):4–7.

DEY, A. K., SALBER, D., ABOWD, G. D. et FUTAKAWA, M. (1999). The conference assistant : Combining context-awareness with wearable computing. Dans *Proceedings of the 3rd IEEE International Symposium on Wearable Computers*, ISWC '99, pages 21–, Washington, DC, USA. IEEE Computer Society.

DINH, Q.-T., EXBRAYAT, M. et VRAIN, C. (2011). Generative structure learning for markov logic networks based on graph of predicates. Dans WALSH, T., éditeur : *IJCAI*, pages 1249–1254. IJCAI/AAAI.

DOCTOR, F., HAGRAS, H. et CALLAGHAN, V. (2005). A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabited environments. *IEEE Transactions on Systems, Man and Cybernetics, Part A : Systems and Humans*, pages 55– 65.

DOUSSON, C., GABORIT, P. et GHALLAB, M. (1993). Situation recognition : Representation and algorithms. Dans *13th International Join Conference on Artificial Intelligence*, pages 166–172, Chambéry, France.

DRUZDZEL, M. J. et LEONG, T.-Y. (2005). Challenges to decision support in a changing world,papers from 2005 aaai spring symposium. Rapport technique SS–05–02, Menlo Park, CA.

DUONG, T., PHUNG, D., BUI, H. et VENKATESH, S. (2009). Efficient duration and hierarchical modeling for human activity recognition. *Artificial Intelligence*, 173(7-8):830–856.

FINE, S., SINGER, Y. et TISHBY, N. (1998). The hierarchical hidden markov model : Analysis and applications. *Machine Learning*, 32:41–62.

FLEURY, A., VACHER, M. et NOURY, N. (2010). SVM-based multi-modal classification of activities of daily living in health smart homes : Sensors, algorithms and first experimental results. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):274 –283.

FLEURY, A., VACHER, M., PORTET, F., CHAHUARA, P. et NOURY, N. (2012). A french corpus of audio and multimodal interactions in a health smart home. *Journal on Multimodal User Interfaces*, pages 1–17. 10.1007/s12193-012-0104-x.

FRIEDEWALD, M., DA COSTA, O., PUNIE, Y., ALAHUHTA, P. et HEINONEN, S. (2005). Perspectives of ambient intelligence in the home environment. *Telemat. Inf.*, 22:221–238.

FRIEDLAND, G., YEO, C. et HUNG, H. (2010). Dialocalizaton : Acoustic speaker diarization and visual localization as joint optimization problem. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 6(4):27 :1–27 :18.

FUSIER, F., VALENTIN, V., BRÉMOND, F., THONNAT, M., BORG, M., THIRDE, D. et FERRYMAN, J. (2007). Video understanding for complex activity recognition. *Machine Vision and Applications*, 18(3-4):167–188.

GACÔGNE, L. (1997). *Eléments de logique floue*. Hermes, Paris.

GALLISSOT, M., CAELEN, J., JAMBON, F. et MEILLON, B. (2013). Une plate-forme usage pour l'intégration de l'informatique ambiante dans l'habitat : Domus. *Technique et Science Informatiques (TSI)*, 32:à paraître.

GAMMA, E., HELM, R., JOHNSON, R. et VLISSIDES, J. (1995). *Design Patterns*. Addison-Wesley, Boston, MA.

GETOOR, L. et TASKAR, B. (2007). *Introduction to Statistical Relational Learning*. The MIT Press.

GIANNOULIS, D., BENETOS, E., STOWELL, D. et PLUMBLEY, M. D. (2012). IEEE AASP CASA challenge - development dataset for event detection task (subtask OL).

GILKS, W., RICHARDSON, S. et SPIEGELHALTER, D. (1995). *Markov Chain Monte Carlo in Practice : Interdisciplinary Statistics*. Chapman and Hall/CRC, 1 édition.

GOC, M. L. (2006). Notion d'observation pour le diagnostic des processus dynamiques : Application à sachem et à la découverte de connaissances temporelles. Mémoire de D.E.A., Université d'Aix-Marseille III.

GRUBER, T. R. (1993). A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220.

GU, T., PUNG, H. K. et ZHANG, D. Q. (2004). A Bayesian Approach for Dealing with Uncertain Contexts. Dans KOTSIS, G., éditeur : *Proceedings of the 2nd International Conference on Pervasive Computing*. Austrian Computer Society.

GU, T., PUNG, H. K. et ZHANG, D. Q. (2005). A service-oriented middleware for building context-aware services. *Journal of Network and Computer Applications*, 28(1):1–18.

HAIGH, K. Z. et YANCO, H. (2002). Automation as caregiver : A survey of issues and technologies. Dans *Proceedings of the AAAI-02 Workshop Automation as Caregiver : The Role of Intelligent Technology in Elder Care*, pages 39–53.

HAMILL, M., YOUNG, V., BOGER, J. et MIHAILIDIS, A. (2009). Development of an automated speech recognition interface for personal emergency response systems. *Journal of NeuroEngineering and Rehabilitation*, 6.

HARTER, A., HOPPER, A., STEGGLES, P., WARD, A. et WEBSTER, P. (1999). The anatomy of a context-aware application. Dans *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, MobiCom '99, pages 59–68, New York, NY, USA. ACM.

HÄUSSERMANN, K., HUBIG, C., LEVI, P., LEYMANN, F., SIMONEIT, O., WIELAND, M. et ZWEIGLE, O. (2010). Understanding and designing situation-aware mobile and ubiquitous computing systems - an interdisciplinary analysis on the recognition of situation with uncertain data using situation templates. Dans *Proceedings of International Conference on Mobile, Ubiquitous and Pervasive Computing*, pages 329–339. WASET.

HAZMAN, M., EL-BELTAGY, S. R. et RAFEA, A. (2011). Article : A survey of ontology learning approaches. *International Journal of Computer Applications*, 22(8):36–43. Published by Foundation of Computer Science.

HENRICKSEN, K., INDULSKA, J. et RAKOTONIRAINY, A. (2002). Modeling context information in pervasive computing systems. Dans *Proceedings of the First International Conference on Pervasive Computing*, Pervasive '02, pages 167–180, London, UK, UK. Springer-Verlag.

HOLMES, A., DUMAN, H. et POUNDS-CORNISH, A. (2002). The idorm : gateway to heterogeneous networking environments. Dans *Proceedings of international ITEA workshop on virtual home environments*, pages 20–37, Paderborn, Germany. ITEA Press.

HOLTZMAN, S. (1989). *Intelligent decision systems*. Addison-Wesley.

HONG, X., NUGENT, C. et MULVENNA, M. (2009). Evidential fusion of sensor data for activity recognition in smart homes. *Pervasive and Mobile Computing*, 5:236–252.

HORRIDGE, M. et BECHHOFER, S. (2011). The owl api : A java api for owl ontologies. *Semantic Web*, 2(1):11–21.

HORROCKS, I., PATEL-SCHNEIDER, P. F., BOLEY, H., TABET, S., GROSOFAND, B. et DEAN, M. (2004). SWRL : A semantic web rule language combining OWL and RuleML. Last access on Dec 2008 at : http ://www.w3.org/Submission/SWRL/.

HOWARD, R. et MATHESON, J. (1981). Influence diagrams. *Readings on The Principles and Applications of Decision Analysis*, 1 and 2:720.

HUANG, W., ZHANG, J. et LIU, Z. (2010). Activity recognition based on hidden markov models. Dans ZHANG, Z. et SIEKMANN, J., éditeurs : *Knowledge Science, Engineering and Management*, volume 4798 de *Lecture Notes in Computer Science*, pages 532–537. Springer Berlin / Heidelberg.

HUYNH, T. N. et MOONEY, R. J. (2008). Discriminative structure and parameter learning for markov logic networks. Dans *Proceedings of the 25th International Conference on Machine Learning (ICML)*, Helsinki, Finland.

INTILLE, S. S. (2002). Designing a home of the future. *IEEE Pervasive Computing*, 1(2):76–82.

ISTRATE, D., CASTELLI, E., VACHER, M., BESACIER, L. et SERIGNAT, J.-F. (2006). Information extraction from sound for medical telemonitoring. *Information Technology in Biomedicine, IEEE Transactions on*, 10(2):264–274.

JAKKULA, V. R., CRANDALL, A. S. et COOK, D. J. (2007). Knowledge discovery in entity based smart environment resident data using temporal relation based data mining. Dans *Proceedings of the Seventh IEEE International Conference on Data Mining Workshops*, pages 625–630. IEEE Computer Society.

JIANMIN JIANG, A. G. et ZHANG, S. (2009). Hermes : A fp7 funded project towards computer-aided memory management via intelligent computations. Dans *3rd Symposium of Ubiquitous Computing and Ambient Intelligence*, pages 249–253.

JR., P. P. M. (1989). Decision making, computer attitudes and expert systems : What is our direction ? *AI Soc.*, 3(2):133–141.

KALDELI, E., WARRIACH, E. U., BRESSER, J., LAZOVIK, A. et AIELLO, M. (2010). Interoperation, composition and simulation of services at home. Dans MAGLIO, P. P., WESKE, M., YANG, J. et FANTINATO, M., éditeurs : *ICSOC*, volume 6470 de *Lecture Notes in Computer Science*, pages 167–181.

KATZ, S. (1983). Assessing self-maintenance : Activities of daily living, mobility, and instrumental activities of daily living. *Journal of the American Geriatrics Society*, 31(12):721–727.

KENT, S. I., LARSON, K. et TAPIA, E. M. (2003). Designing and evaluating technology for independent aging in the home. Dans *In International Conference on Aging, Disability and Independence*.

KERSTING, K., RAEDT, L. D. et RAIKO, T. (2006). Logical hidden markov models. *Journal of Artificial Intelligence Research*, 25:425–456.

KHATTAK, A., VINH, L. T., HUNG, D. V., TRUC, P. T. H., HUNG, L. X., GUAN, D., PERVEZ, Z., HAN, M., LEE, S. et LEE, Y.-K. (2010). Context-aware human activity recognition and decision making. Dans *2010 12th IEEE International Conference on e-Health Networking Applications and Services (Healthcom)*, pages 112 –118.

KINDERMANN, R. (1980). *Markov Random Fields and Their Applications (Contemporary Mathematics ; V. 1)*. American Mathematical Society.

KOK, S. et DOMINGOS, P. (2005). Learning the structure of markov logic networks. Dans *Proceedings of the 22nd international conference on Machine learning*, ICML '05, page 441–448, New York, NY, USA. ACM.

KOK, S. et DOMINGOS, P. (2010). Learning markov logic networks using structural motifs. Dans FÜRNKRANZ, J. et JOACHIMS, T., éditeurs : *ICML*, pages 551–558. Omnipress.

KOLOVSKI, V., PARSIA, B. et SIRIN, E. (2006). Extending the *shoiq(d)* tableaux with dl-safe rules : First results. Dans *Description Logics*.

KOWALSKI, R. A. et SERGOT, M. J. (1986). A logic-based calculus of events. *New Generation Comput.*, 4(1):67–95.

KRÖSE, B., van KASTEREN, T., GIBSON, C. et van den DOOL, T. (2008). Care : Context awareness in residences for elderly. Dans *The 6th International Conference of the International Society of Gerontology*, Pisa, Italy.

KWON, O., YOO, K. et SUH, E. (2005). Ubidss : a proactive intelligent decision support system as an expert system deploying ubiquitous computing technologies. *Expert Syst. Appl.*, 28(1):149–161.

LACEY, G. et MACNAMARA, S. (2000). Context-aware shared control of a robot mobility aid for the elderly blind. *I. J. Robotic Res.*, 19(11):1054–1065.

LACOMBE, A., ROCARIES, F., DIETRICH, C., BALDINGER, J., BOUDY, J., DELAVAULT, F., DESCATHA, A., BAER, M. et OZGULER, A. (2005). Open technical platform prototype and validation process model for patient at home medical monitoring system. Dans *BioMedsim*, Linköping, Sweden.

LE BELLEGO, G., NOURY, N., VIRONE, G., MOUSSEAU, M. et DEMONGEOT, J. (2006). A model for the measurement of patient activity in a hospital suite. *IEEE Transactions on Information Technologies in Biomedicine*, 10(1):92 – 99.

LEE, S.-H. et CHO, S.-B. (2012). Fusion of modular bayesian networks for context-aware decision making. Dans CORCHADO, E., SNÁŠEL, V., ABRAHAM, A., WOZNIAK, M., GRAÑA, M. et CHO, S.-B., éditeurs : *Hybrid Artificial Intelligent Systems*, volume 7208 de *Lecture Notes in Computer Science*, pages 375–384. Springer Berlin / Heidelberg.

LI, S. et WREDE, B. (2007). Why and how to model multi-modal interaction for a mobile robot companion. Dans *AAAI Spring Symposium 2007 on Interaction Challenges for Intelligent Assistants*.

LIAO, H.-C. et TU, C.-C. (2007). A rdf and owl-based temporal context reasoning model for smart home. *Information Technology Journal*, 6(8):1130–1138.

LINES, L. et HONE, K. S. (2006). Multiple voices, multiple choices : Older adults' evaluation of speech output to support independent living. *Gerontechnology Journal*, 5(2):78–91.

LITVAK, D., ZIGEL, Y. et GANNOT, I. (2008). Fall detection of elderly through floor vibrations and sound. Dans *Proc. 30th Annual Int. Conference of the IEEE-EMBS 2008*, pages 4632–4635.

LOWD, D. et DOMINGOS, P. (2007). Efficient weight learning for markov logic networks. Dans *In Proceedings of the Eleventh European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 200–211.

LÓPEZ-CÓZAR, R. et CALLEJAS, Z. (2010). Multimodal dialogue for ambient intelligence and smart environments. Dans NAKASHIMA, H., AGHAJAN, H. et AUGUSTO, J. C., éditeurs : *Handbook of Ambient Intelligence and Smart Environments*, pages 559–579. Springer US.

MAIER, E. et KEMPTER, G. (2010). ALADIN – a magic lamp for the elderly ? Dans *Handbook of Ambient Intelligence and Smart Environments*, pages 1201–1227.

MCCARTHY, J. (1993). Notes on formalizing context. Dans *Proceedings of the 13th international joint conference on Artifical intelligence - Volume 1*, IJCAI'93, pages 555–560, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

MEDJAHED, H., ISTRATE, D., BOUDY, J. et DORIZZI, B. (2009). A fuzzy logic system for home elderly people monitoring (EMUTEM). Dans *Proceedings of the 10th WSEAS international conference on Fuzzy systems*, pages 69–75, Prague, Czech Republic. World Scientific and Engineering Academy and Society (WSEAS).

MENDEL, J. M. (1995). Fuzzy logic system for engineering : a tutorial. *Proceeding of the IEEE*, 83:984–993.

MICERA, S., BONATO, P. et TAMURA, T. (2008). Gerontechnology. *IEEE Engineering in Medicine and Biology Magazine*, 27(4):10 –14.

MILCH, B., MARTHI, B., RUSSELL, S. J., SONTAG, D., ONG, D. L. et KOLOBOV, A. (2005). Blog : Probabilistic models with unknown objects. Dans *IJCAI*, pages 1352–1359.

MILEO, A., MERICO, D. et BISIANI, R. (2010). Support for context-aware monitoring in home healthcare. *J. Ambient Intell. Smart Environ.*, 2(1):49–66.

MILHORAT, P., ISTRATE, D., BOUDY, J. et CHOLLET, G. (2012). Hands-free speech-sound interactions at home. Dans *Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012*, pages 1678 –1682.

MITRA, K., ZASLAVSKY, A. B. et ÅHLUND, C. (2011). A probabilistic context-aware approach for quality of experience measurement in pervasive systems. Dans *SAC*, pages 419–424.

MOORE, P., HU, B. et JACKSON, M. (2011). Rule strategies for intelligent context-aware systems : The application of conditional relationships in decision-support. Dans *International Conference on Complex, Intelligent and Software Intensive Systems, CISIS 2011*, pages 9–16, Seoul, Korea.

MOZER, M. C. (2005). *Smart environments : Technologies, protocols, and applications*. J. Wiley & Sons.

MUGGLETON, S. (1995). Inverse entailment and progol. *New Generation Computing*, 13(3): 245–286.

MUGGLETON, S. et RAEDT, L. D. (1994). Inductive logic programming : Theory and methods. *Journal of Logic Programming*, 19(20):629—679.

NAEEM, U. et BIGHAM, J. (2007). A comparison of two hidden markov approaches to task identification in the home environment. Dans *Pervasive Computing and Applications, 2007. ICPCA 2007. 2nd International Conference on*, pages 383–388.

NAEEM, U. et BIGHAM, J. (2008). Activity recognition using a hierarchical framework. *Pervasive Computing Technologies for Healthcare*, pages 24 – 27.

NATARAJAN, S., BUI, H. H., TADEPALLI, P., KERSTING, K. et WONG, W. (2008). Logical hierarchical hidden markov models for modeling user activities. Dans *Proceedings of the 18th international conference on Inductive Logic Programming*, pages 192–209, Prague, Czech Republic. Springer-Verlag.

NATH, A. et DOMINGOS, P. (2009). A language for relational decision theory. Dans *Proceedings of the Sixth International Workshop on Statistical Relational Learning*, Leuven, Belgium.

NATH, A. et DOMINGOS, P. (2010). Efficient belief propagation for utility maximization and repeated inference. Dans *AAAI*.

NEUMANN, J. V. et MORGENSTERN, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press.

NIC, U. H. L. M. M. S. (2003). *Pervasive Computing : The Mobile World*. Springer, 2nd édition.

NIESSEN, M. E., van MAANEN, L. et ANDRINGA, T. C. (2008). Disambiguating sounds through context. Dans *Proceedings of the 2008 IEEE International Conference on Semantic Computing*, pages 88–95. IEEE Computer Society.

NISHIYAMA, T., HIBIYA, S. et SAWARAGI, T. (2011). Development of agent system based on decision model for creating an ambient space. *AI & Society*, 26(3):247–259.

O'LEARY, D. et KANDELIN, N. A. (1988). Validating the weights in rule-based expert systems : a statistical approach. *Int. J. Expert Syst.*, 1(3):253–279.

PARLAK, S., MARSIC, I. et BURD, R. S. (2011). Activity recognition for emergency care using rfid. Dans *Proceedings of the 6th International Conference on Body Area Networks*, BodyNets '11, pages 40–46, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

PARSIA, B., SIRIN, E. et KALYANPUR, A. (2005). Debugging owl ontologies. Dans *Proc. 14th Intel Conf. World Wide Web*, pages 633–640. ACM Press.

PATTERSON, D., ETZIONI, O., FOX, D. et KAUTZ, H. (2002). Intelligent ubiquitous computing to support alzheimer's patients : Enabling the cognitively disabled. Dans *UbiCog '02 : First International Workshop on Ubiquitous Computing for Cognitive Aids*, Göteborg, Sweden.

PEARL, J. (1982). Reverend bayes on inference engines : A distributed hierarchical approach. Dans *AAAI*, pages 133–136.

POMPONIO, L., GOC, M. L., PASCUAL, E. et ANFOSSO, A. (2011). Combining timed data and expert's knowledge to model human behavior. Dans *Proceedings of the Health Ambiant Information Systems Workshop, HamIS 2011. Co-located with the 6th Ibero-american Congress on Telematics, CITA 2011*, volume 729, Gramado, Brazil.

POON, H. et DOMINGOS, P. (2006). Sound and efficient inference with probabilistic and deterministic dependencies. Dans *AAAI*, pages 458–463.

POPESCU, M., LI, Y., SKUBIC, M. et RANTZ, M. (2008). An acoustic fall detector system that uses sound height information to reduce the false alarm rate. Dans *Proc. 30th Annual Int. Conference of the IEEE-EMBS 2008*, pages 4628–4631.

PORTET, F., FLEURY, A., VACHER, M. et NOURY, N. (2009). Determining useful sensors for automatic recognition of activities of daily living in health smart home. Dans *in Intelligent Data International Workshop on Analysis in Medicine and Pharmacology (IDAMAP2009)*, pages 63–64, Verona, Italy.

PORTET, F., VACHER, M., GOLANSKI, C., ROUX, C. et MEILLON, B. (2013). Design and evaluation of a smart home voice interface for the elderly : acceptability and objection aspects. *Personal and Ubiquitous Computing*, 17:127–144.

RAAD, M. W. et YANG, L. T. (2009). A ubiquitous smart home for elderly. *Information Systems Frontiers*, 11(5):529–536.

RABINER, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

RAEDT, L. D., KIMMIG, A. et TOIVONEN, H. (2007). Problog : A probabilistic prolog and its application in link discovery. Dans *IJCAI*, pages 2462–2467.

RAIFFA, H. (1970). *Decision Analysis - Introductory lectures on choices under uncertainty*. Addison Wesley, Reading, MA. traduction française : Analyse de la décision : introduction aux choix en avenir incertain, Dunod, 1973.

RAMOS, C., AUGUSTO, J. C. et SHAPIRO, D. (2008). Ambient Intelligence ;the Next Step for Artificial Intelligence. *IEEE Intelligent Systems*, 23(2):15–18.

RANTZ, M., PORTER, R., CHESHIER, D., OTTO, D., SERVEY, C., JOHNSON, R., AUD, M., SKUBIC, M., TYRER, H., HE, Z., DEMIRIS, G., ALEXANDER, G. et TAYLOR, G. (2008). TigerPlace, a State-Academic-Private project to revolutionize traditional Long-Term care. *Journal of Housing For the Elderly*, 22(1):66.

RASCH, K., LI, F., SEHIC, S., AYANI, R. et DUSTDAR, S. (2011). Context-driven personalized service discovery in pervasive environments. *World Wide Web*, 14(4):295–319.

RASHIDI, P., COOK, D. J., HOLDER, L. B. et SCHMITTER-EDGECOMBE, M. (2011). Discovering activities to recognize and track in a smart environment. *IEEE Trans. Knowl. Data Eng.*, 23(4):527–539.

RASMY, M. H., LEE, S. M., EL-WAHED, W. F. A., RAGAB, A. M. et EL-SHERBINY, M. M. (2002). An expert system for multiobjective decision making : application of fuzzy linguistic preferences and goal programming. *Fuzzy Sets and Systems*, 127(2):209–220.

RIALLE, V. (2007). Rapport sur les technologies nouvelles susceptibles d'améliorer les pratiques gérontologiques et la vie quotidienne des malades âgés et de leur famille. Rapport technique, Université Joseph Fourier et CHU de Grenoble.

RICHARDSON, M. et DOMINGOS, P. (2006). Markov logic networks. *Machine Learning*, 62(1-2):107–136.

RICQUEBOURG, V., DURAND, D., MENGA, D., MARHIC, B., DELAHOCHE, L., LOGÉ, C. et JOLLY-DESODT, A.-M. (2007). Context inferring in the smart home : An swrl approach. Dans *AINA Workshops (2)*, pages 290–295.

ROMDHANE, R., MULIN, E., DERREUMEAUX, A., ZOUBA, N., PIANO, J., LEE, L., MALLEA, P., THONNAT, M., BREMOND, F., ROBERT, P., LEROI, I. et DAVID, R. (2011). Automatic video monitoring system for assessment of alzheimer's disease symptoms. *The Journal of Nutrition, Health and Aging*, 16(3).

RONZANI, D. (2009). The battle of concepts : Ubiquitous computing, pervasive computing and ambient intelligence in mass media. *UbiCC Journal*, 4(2).

ROTA, N. et THONNAT, M. (2000). Activity recognition from video sequences using declarative models. Dans HORN, W., éditeur : *ECAI*, pages 673–680. IOS Press.

ROUGUI, J., ISTRATE, D. et SOUIDENE, W. (2009). Audio sound event identification for distress situations and context awareness. Dans *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 3501–3504, Minneapolis, USA.

RUAN, D., MUÑOZ, A., BOTÍA, J. A. et AUGUSTO, J. C. (2010). Intelligent decision-making for a smart home environment with multiple occupants. Dans RUAN, D., éditeur : *Computational Intelligence in Complex Decision Systems*, volume 2, pages 325–371. Atlantis Press, Paris.

RUMEAU, P., RIALLE, V. et NOURY, N. (2006). A priori evaluation of acceptance of an activity monitoring device for the disabled elderly people using the HIS as a model. Dans *Proceedings of the 4th International Conference on Smart Homes and Health Telematics, ICOST2006, Smart Homes and Beyond*, pages 130–137, Belfast, Ireland. IOS Press.

RUSSELL, S. J. et NORVIG, P. (2003). *Artificial Intelligence : Modern Approach*. Prentice Hall, 2 édition.

RYAN, N., PASCOE, J. et MORSE, D. (1997). Enhanced reality fieldwork : the context-aware archaeological assistant. Dans GAFFNEY, V., van LEUSEN, M. et EXXON, S., éditeurs : *Computer Applications and Quantitative Methods in Archaeology (CAA 97), Oxford*.

SCHIELE, G., HANDTE, M. et BECKER, C. (2010). Pervasive computing middleware. Dans *Handbook of Ambient Intelligence and Smart Environments*, pages 201–227.

SCHILIT, B., ADAMS, N. et WANT, R. (1994). Context-aware computing applications. Dans *In Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society.

SCHILIT, B. N. et THEIMER, M. M. (1994). Disseminating active map information to mobile hosts. *Network, IEEE*, 8(5):22–32.

SCHMIDT, A., AIDOO, K. A., TAKALUOMA, A., TUOMELA, U., LAERHOVEN, K. V. et VELDE, W. V. d. (1999). Advanced interaction in context. Dans *Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, HUC '99, pages 89–101, London, UK, UK. Springer-Verlag.

SHACHTER, R. D. (1986). Evaluating Influence Diagrams. *Operations Research*, 34(6):871–882.

SIMON et DILLON, S. M. (1960). The new science of management decision. Dans *In Proceedings of the 33 rd Conference of the Operational Research Society of New Zealand*.

SINGH, V., GAO, M. et JAIN, R. (2012). Situation recognition : An evolving problem for heterogeneous dynamic big multimedia data. Dans *ACM Multimedia (ACM MM '12)*, Nara, Japan.

SINGLA, G., COOK, D. et SCHMITTER-EDGECOMBE, M. (2008). Incorporating temporal reasoning into activity recognition for smart home residents. Dans *AAAI Workshop on Spatial and Temporal Reasoning*, pages 53–61.

SINGLA, P. et DOMINGOS, P. (2005). Discriminative training of markov logic networks. Dans *Proceedings of the 20th national conference on Artificial intelligence - Volume 2*, AAAI'05, pages 868–873. AAAI Press.

SKARLATIDIS, A., PALIOURAS, G., VOUROS, G. A. et ARTIKIS, A. (2011). Probabilistic event calculus based on markov logic networks. Dans *RuleML America*, volume 7018 de *Lecture Notes in Computer Science*, pages 155–170. Springer.

SKUBIC, M., ALEXANDER, G., POPESCU, M., RANTZ, M. et KELLER, J. (2009). A smart home application to eldercare : Current status and lessons learned. *Technology and Health Care*, 17(3):183–201.

SOYLU, A., CAUSMAECKER, P. et DESMET, P. (2009). Context and adaptivity in pervasive computing environments : Links with software engineering and ontological engineering. *Journal of Software*, 4(9):992–1013.

SPECTOR, B. I. (1993). Decision analysis for practical negotiation application. *Theory and Decision*, 34(3):183–199.

STEUER, R. E. (1986). *Multiple Criteria Optimization : Theory, Computations, and Application.* John Wiley & Sons, Inc.

STIKIC, M., HUYNH, T., LAERHOVEN, K. V. et SCHIELE, B. (2008). Adl recognition based on the combination of rfid and accelerometer sensing. Dans *Pervasive Computing Technologies for Healthcare*, pages 258–263, Tampere, Finland.

STORF, H., BECKER, M. et RIEDL, M. (2009). Rule-based activity recognition framework : Challenges, technique and learning. Dans *Pervasive Computing Technologies for Healthcare*, pages 1–7, London, UK.

STRICKER, MARTIN et GROSS (2010). Realtime user attention and emotion estimation on a mobile robot. Dans *Ilmenau International Scientific Colloquium*, Ilmenau, Germany.

TANG, P. et VENABLES, T. (2000). Smart homes and telecare for independent living. *Journal of Telemedicine and Telecare*, 6(1):8–14.

TAPIA, E. M., INTILLE, S. S. et LARSON, K. (2004). Activity recognition in the home using simple and ubiquitous sensors. *Pervasive Computing*, 2:158–175.

TAWIL, E. et HAGRAS, H. (2005). Adaptive on-line co-ordination of ubiquitous computing devices with multiple objectives and constraints. *The IEE International Workshop on Intelligent Environments*, (11059):116–124.

TRAN, S. D. et DAVIS, L. S. (2008). Event modeling and recognition using markov logic networks. Dans *Proceedings of the 10th European Conference on Computer Vision : Part II*, pages 610–623, Marseille, France. Springer-Verlag.

VACHER, M., FLEURY, A., PORTET, F., SERIGNAT, J.-F. et NOURY, N. (2010a). *Complete Sound and Speech Recognition System for Health Smart Homes : Application to the Recognition of Activities of Daily Living*, pages 645 – 673. Intech Book.

VACHER, M., ISTRATE, D., PORTET, F., JOUBERT, T., CHEVALIER, T., SMIDTAS, S., MEILLON, B., LECOUTEUX, B., SEHILI, M., CHAHUARA, P. et MÉNIARD, S. (2011a). The sweet-home project : Audio technology in smart homes to improve well-being and reliance. Dans *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011)*, Boston, USA.

VACHER, M., PORTET, F., FLEURY, A. et NOURY, N. (2010b). Challenges in the processing of audio channels for ambient assisted living. Dans *IEEE HealthCom 2010*, pages 330–337.

VACHER, M., PORTET, F., FLEURY, A. et NOURY, N. (2011b). Development of audio sensing technology for ambient assisted living : Applications and challenges. *International Journal of E-Health and Medical Communications*, 2(1):35–54.

van KASTEREN, T. et KROSE, B. (2007). Bayesian activity recognition in residence for elders. Dans *3rd IET International Conference on Intelligent Environments*, pages 209–212, Ulm, Germany.

VAUFREYDAZ, D., BERGAMINI, C., SERIGNAT, J.-F., BESACIER, L. et AKBAR, M. (2000). A new methodology for speech corpora definition from Internet documents. Dans *Proc. LREC'2000, 2nd Int. Conf. on Language Ressources and Evaluation*, pages 423–426, Athens, Greece.

WANG, F., SKUBIC, M., ABBOTT, C. et KELLER, J. (2011). Quantitative analysis of 180 degree turns for fall risk assessment using video sensors. *Conf Proc IEEE Eng Med Biol Soc*, 2011.

WARD, J. A., LUKOWICZ, P., TROSTER, G. et STARNER, T. E. (2006). Activity Recognition of Assembly Tasks Using Body-Worn Microphones and Accelerometers. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1553–1567.

WEI, E. J. Y. et CHAN, A. T. S. (2010). Towards semantic-based adaptation decisions for context-aware mobile computing. Dans *SAC*, pages 563–567.

WEISER, M. (1994). The world is not a desktop. *ACM Interactions*, 1(1):7–8.

WILSON, D., WYATT, D. et PHILIPOSE, M. (2005). Using context history for data collection in the home. Dans *Proceedings of Pervasive 2005*, Munich, Germany.

WOLF, P., SCHMIDT, A. et KLEIN, M. (2008). Soprano - an extensible, open aal platform for elderly people based on semantical contracts. Dans *3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI'08), 18th European Conference on Artificial Intelligence (ECAI 08), Patras, Greece*.

XI, C., ZHIHAI, H., KELLER, J. M., ANDERSON, D. et SKUBIC, M. (2006). Adaptive silhouette extraction in dynamic environments using fuzzy logic. Dans *International Conference on Fuzzy Systems*, pages 236–243, Vancouver, Canada.

XU, Y., STOJANOVIC, L. et andDarko ANICIC, J. M. (2011). Efficient energy consumption in a smart office based on intelligent complex event processing. Dans *Proceedings of the Fifth ACM International Conference on Distributed Event-Based Systems*, pages 379–380, New York, USA.

XU, Y., WOLF, P., STOJANOVIC, N. et HAPPEL, H.-J. (2010). Semantic-based complex event processing in the aal domain. Dans *International Semantic Web Conference 2010*, Shanghai,China.

YAN, H. et SELKER, T. (2000). Context-aware office assistant. Dans *Proceedings of the 5th international conference on Intelligent user interfaces*, IUI '00, pages 276–279, New York, NY, USA. ACM.

YANG, Y. et CALMET, J. (2006). From the ontobayes model to a service oriented decision support system. Dans *CIMCA/IAWTIC*, page 127.

YE, J., DOBSON, S. et MCKEEVER, S. (2012). Review : Situation identification techniques in pervasive computing : A review. *Pervasive Mob. Comput.*, 8(1):36–66.

YILMAZ, O. et ERDUR, R. C. (2012). iConAwa – an intelligent context-aware system. *Expert Systems with Applications*, 39(3):2907–2918.

YU, S.-Z. (2010). Hidden semi-markov models. *Artificial Intelligence*, 174(2):215–243.

ZADEH, L. (1965). Fuzzy sets. *Information and Control*, 8(3):338–353.

ZELKHA, E. (1998). The future of information appliances and consumer devices. Palo Alto Ventures.

ZHANG, N. et POOLE, D. (1992). Stepwise decomposable influence diagrams. Dans *Proceedings of the Fourth International Conference on Knowledge Representation and Reasoning*, pages 141–152.

ZOUBA, N., BREMOND, F., THONNAT, M., ANFOSSO, A., PASCUAL, E., MALLEA, P., MAILLAND, V. et GUERIN, O. (2009). A computer system to monitor older adults at home : Preliminary results. *Gerontechnology Journal*, 8(3):129–139.

Bibliographie personelle

[1] Anthony Fleury, Michel Vacher, François Portet, Pedro Chahuara and Norbert Noury. A French corpus of audio and multimodal interactions in a health smart home.*Journal on Multimodal User Interfaces*. 2012.

[2] Pedro Chahuara, Anthony Fleury, François Portet, Michel Vacher.Using Markov Logic Network for On-line Activity Recognition from Non-Visual Home Automation Sensors. *International Joint Conference on Ambient Intelligence*.2012.

[3] Pedro Chahuara, François Portet, Michel Vacher. Context aware decision system in a smart home : knowledge representation and decision making using uncertain contextual information. *The 4th International Workshop on Acquisition, Representation and Reasoning with Contextualized Knowledge (ARCOE-12)*.2012.

[4] Pedro Chahuara, Anthony Fleury, Michel Vacher, François Portet. Méthodes SVM et MLN pour la reconnaissance automatique d'activités humaines dans les habitats perceptifs : tests et perspectives. *Reconnaissance des Formes et l'Intelligence Artificielle (RFIA)*, Lyon, France,2012.

[5] Pedro Chahuara, Michel Vacher, François Portet. Localisation d'habitant dans un espace perceptif par réseau dynamique, *Représentation et Raisonnement sur le Temps et l'Espace (RTE)*, Chambéry, France, 2011.

[6] Pedro Chahuara, François Portet, Michel Vacher. Location of an Inhabitant for Domotic Assistance Through Fusion of Audio and Non-Visual Data, *Pervasive Health, Dublin, Ireland*,2011.

[7] Michel Vacher, Dan Istrate, François Portet, Thierry Joubert, Thierry Chevalier, Serge Smidtas, Brigitte Meillon, Benjamin Lecouteux, Mohamed Sehili, Pedro Chahuara, Sylvain Meniard. The SWEET-HOME Project : Audio Technology in Smart Homes to improve Well-being and Reliance, *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2011),Boston, USA*, 2011.

[8] Pedro Chahuara, François Portet, Michel Vacher. Fusion of Audio and Temporal Multimodal Data by Spreading Activation for Dweller Localisation in a Smart Home, *STAMI Series, Space, Time and Ambient Intelligence*,2011

[9] Anthony Fleury, Michel Vacher, François Portet, Pedro Chahuara, Norbert Noury. A Multimodal Corpus Recorded in a Health Smart Home, *LREC Workshop Multimodal Corpora and Evaluation*, Malta, 2010.

[10] Pedro Chahuara, Michel Vacher, François Portet. Localisation d'habitant dans un environnement perceptif non visuel par propagation d'activations multisource, *MajecSTIC*, Bordeaux, France, 2010.

Index

Commande domotique, 60, 133
Contexte, 38
Contrôleur intelligent, 57, 62

Diagramme d'influence, 121, 122
Domotique, 26

Environnement intelligent, 24

Informatique ubiquitaire, 23
Intelligence ambiante, 24

KNX, 128

Localisation, 41, 43, 65, 103

Maison intelligente, 27

Ontologie, 69, 91

Plateforme
    DOMUS, 128, 129, 136, 138, 139
    HIS-TIMC, 136, 138, 139
Prise de décision, 49, 115
Projet
    Aging in Place, 33
    CASAS, 31, 38
    Companionable, 34
    GERHOME, 32, 38
    HomeCare, 36
    House n, 30, 38
    iSpace, 31, 38
    SM4ALL, 35
    Soprano, 34
    SWEET HOME, 58
Propagation de l'activation, 105

Réseau Bayésien, 121, 122
Réseau logique de Markov, 97, 107, 122
Réseaux Bayésien, 76
Règles logiques, 86, 94

Risque, 116
RSB, 104, 137

Scénarios, 61
Situation, 39, 40, 112
SVM, 76
SWRL, 94, 143

Utilité, 116

ANNEXE A

Scénario de l'expérimentation 1 : constitution d'un corpus multimodal

Actions à réaliser	Durée	Consignes
Début		
Entrer dans l'appartement et fermer la porte Actionner la gâche électrique Faire le tour des pièces en fermant les portes derrière soit Poser le manteau ou la veste Revenir à la cuisine	~ 30 secondes	vous pouvez allumer la lumière si il fait trop sombre
Habillage		
Aller dans la chambre et fermer la porte derrière soi Sortir les vêtements du tiroir (tee-shirt et pantalon) Enfiler tee-shirt et pantalon par desssus ses vêtements Ouvrir les volets et les fenêtres de la chambre	~ 3 minutes	vous pouvez allumer la lumière si il fait trop sombre
Hygiène		
Aller dans la salle de bain, éclairer la lumière Se laver les mains, s'essuyer, éteindre la lumière	~ 3 minutes	
Manger		
Aller à la cuisine Ouvrir le volet de la cuisine Préparer les ustensiles de cuisine pour faire une boisson chaude	~ 15 minutes	

Voir en page suivante

Actions à réaliser	**Durée**	**Consignes**
Préparer une boisson chaude en utilisant une casserole Prendre une tasse du placard et du pain, confitures, fruits, etc. S'assoir et manger sur la table de la cuisine Débarrasser la table Nettoyer la table Faire la vaisselle		thé, café, tisane, chocolat
Non disponibilité lors d'un appel extérieur		
Recevoir un appel téléphonique pendant la vaisselle et ne pas répondre		
Faire la vaisselle		
Essuyer la vaisselle Ranger la vaisselle		
Hygiène - se laver les dents après le repas		
Aller dans la salle de bain, allumer la lumière Tousser Se racler la gorge Se brosser les dents Cracher de l'eau Se rincer la bouche Boire de l'eau Claquer les mains Éteindre la lumière avant de sortir	~ 3 minutes	
Sieste		
Aller dans la chambre Fermer les portes Fermer les volets et les fenêtre de la chambre Éteindre tous les lumières en restant dans la chambre Se coucher Dormir au moins 1 minute Se lever et aller à la cuisine pour se servir un verre d'eau	~ 10 minutes	

Voir en page suivante

Actions à réaliser	Durée	Consignes
Se recoucher Dormir au moins 2 minutes		
Se lever		
Se lever et allumer la lumière Ouvrir les fenêtres et les volets Faire le lit	~ 10 minutes	
Ménage		
Prendre l'aspirateur dans le placard de la cuisine Passer l'aspirateur dans la cuisine et la chambre Aller dans la cuisine Sortir les assiettes une à une et les poser sur la table Prendre tous les couverts et les poser sur la table Prendre les produits de nettoyage dans le placard de la cuisine Laver l'évier de la cuisine Fermer les fenêtres Ranger la vaisselle dans le placard	~ 10 minutes	
Détente		
Aller dans le bureau et chercher un livre sur l'étagère Lire le livre dans le bureau Rester dans le bureau, mettre en route la radio puis écouter Aller à la cuisine pour prendre une boisson froide Continuer à lire dans le bureau	~ 10 minutes	verre d'eau
Appel téléphonique		
Recevoir un appel, éteindre la télé/radio et avoir une conversation téléphonique	~ 5 minutes	lire le texte 1, respecter une pause entre chaque phrase
Détente		

Voir en page suivante

Actions à réaliser	**Durée**	**Consignes**
Remettre en route la radio Ecouter la radio Éteindre la radio	~ 5 minutes	
Sortie		
Fermer tous les fenêtres Éteindre toutes les lumières Prendre son manteau ou sa veste Sortir de l'appartement et fermer à clef	~ 3 minutes	
Retour de sortie		
Ouvrir avec la clef Entrer dans l'appartement Poser son manteau ou son pull dans la chambre Ranger les courses dans la cuisine	~ 3 minutes	
Détente/communication		
Aller dans le bureau Allumer l'ordinateur Utiliser l'ordinateur pour consulter le web Éteindre l'ordinateur	~ 10 minutes	
Communication téléphonique		
Effectuer un appel téléphonique	~ 5 minutes	lire le texte 2, respecter une pause entre chaque phrase
Remplir le sac de courses		
Aller à la cuisine Boire un verre d'eau Remettre les courses dans le sac		opération inverse de la précédente
Déshabillage		
Aller dans la chambre Allumer le radiateur Se déshabiller	~ 3 minutes	
Hygiène		
Prendre une douche	~ 5 minutes	faire couler la douche et simuler

Voir en page suivante

Actions à réaliser	**Durée**	**Consignes**
Éteindre la lumière Sortir de la salle de bain en fermant la porte		aller dans la chambre
Habillage		
S'habiller pour la nuit	~ 3 minutes	
Vérification des issues		
Vérifier les volets, les fenêtres Verrouiller la porte d'entrée éteindre le radiateur	~ 3 minutes	
Se coucher pour la nuit		
Se coucher Lire quelques pages d'un livre Éteindre la lumière Dormir	~ 10 minutes	

ANNEXE B

Scénario de l'expérimentation 2 : évaluation du contrôleur intelligent

Actions à réaliser
Vous souhaitez vous préparer une collation
Allez à la cuisine Demandez l'ouverture des stores de la cuisine (commande vocale) Rassemblez les ustensiles et ingrédients pour faire une boisson chaude Demandez d'allumer la lumière (commande vocale) Préparez une boisson chaude (thé ou tisane) en utilisant une casserole et la plaque électrique Sortez du placard une tasse et des petits gâteaux (si pas déjà fait)
Vous souhaitez maintenant prendre votre collation
Asseyez-vous à la table de la cuisine Demandez d'allumer la lumière (commande vocale) Vous pouvez maintenant boire la boisson et manger quelques gâteaux
Vous souhaitez faire la vaisselle
Débarrassez la table Lavez la vaisselle Nettoyez la table Demandez l'heure (commande vocale) Eteignez la lumière (commande vocale ou interrupteur) Allez dans la chambre Allumez la lumière (commande vocale ou interrupteur) Demandez la température (commande vocale) Entrouvrez une fenêtre de la chambre et laissez la ouverte
Vous vous êtes aperçu(e) que le placard sous l'évier est sale et vous souhaitez le nettoyer
Allez dans la cuisine Sortez les couverts du placard et posez-les sur la tablehead
Voir en page suivante

Actions à réaliser
Sortez les assiettes du placard une par une et posez-les sur la tablehead
Demandez d'allumer la lumière (commande vocale)
Prenez le produit de nettoyage dans le placard de la cuisine
Nettoyez le placard
Lavez l'évier de la cuisine
Rangez la vaisselle dans le placard
Vous souhaitez aller faire des courses
Eteignez toutes les lumières de l'appartement (interrupteurs)
Sortez de l'appartement et fermez la porte à clé
Prenez le sac de courses dans le couloir
Restez dehors quelques minutes
Entrez dans l'appartement et laissez la porte ouverte
Demandez l'heure (commande vocale)
Vous vous rendez compte qu'il est tard :
Fermez les stores et allumez la lumière (commande vocale ou interrupteur)
puis rangez les courses dans le placard
Se déplacer dans DOMUS
Allez dans le salon
Prenez un livre
Demandez la fermeture du store (commande vocale)
Demandez d'allumer la lumière (commande vocale)
Installez-vous dans un fauteuil
Lisez
Simulez un appel de détresse : "Nestor, aide moi" "Nestor, appelle mon fils" (commande vocale)
Vous souhaitez faire la sieste
Eteignez la lumière (commande vocale ou interrupteurs)
Allez dans la chambre
Fermez les portes
Demandez de fermer les stores et les rideaux (commande vocale)
Allumez la lumière (commande vocale ou interrupteurs)
Asseyez-vous sur le lit
Demandez d'éteindre la lumière (commande vocale)
Allez fermer la porte d'entrée
Retournez dans la chambre et allongez-vous
Reposez-vous au moins 2 minutes
Allumez la lumière (commande vocale)
Voir en page suivante

Actions à réaliser
Levez-vous
Demandez d'allumer la radio (commande vocale)
Allez dans le salon
Demandez d'éteindre la radio (commande vocale)
Vous pouvez sortir de l'appartement