Etude de l'émergence de facultés d'apprentissage fiables et prédictibles d'actions réflexes, à partir de modèles paramétriques soumis à des contraintes internes.

Frédéric Davesne

Laboratoire
Systèmes Complexes

Centre d'Études Mécaniques
d'Ile-de-France

Étude de l'émergence de facultés d'apprentissage fiables et prédictibles d'actions réflexes, à partir de modèles paramétriques soumis à des contraintes internes

THÈSE

présentée et soutenue publiquement le 19 Avril 2002

pour l'obtention du

Doctorat de l'Université d'Évry Val d'Essonne
(spécialité Génie Informatique et Robotique)

par

Frédéric Davesne

Composition du jury

Président :	M. Jeanny Hérault (Pr. INPG, Grenoble)
Rapporteurs :	M. Florent Chavand (Pr. Univ. Evry) M. Philippe Gaussier (Pr. Univ. Cergy Pontoise) M. Pierre-Yves Glorennec (Pr. INSA Rennes)
Examinateur :	M. Claude Barret (Pr. Univ. Evry)

Mis en page avec la classe thloria.

Résumé

L'objectif à long terme de notre travail est la mise au points de techniques d'apprentissage **fiables et prédictibles** d'actions réflexes, dans le cadre de la robotique mobile. Ce document constitue un départ à ce projet.

Dans un premier temps, nous donnons des arguments défendant l'idée que les méthodes d'apprentissage classiques ne peuvent pas, intrinsèquement, répondre à nos exigences de fiabilité et de prédictibilité. Nous pensons que **la clé du problème se situe dans la manière dont la communication entre le système apprenant et son environnement est modélisée**. Nous illustrons nos propos grâce à un exemple d'apprentissage par renforcement.

Nous présentons une démarche **formalisée** dans laquelle la communication est une **interaction**, au sens physique du terme. Le système y est soumis à deux forces : la réaction du système est due à la fois à l'action de l'environnement et au **maintient de contraintes internes**. **L'apprentissage devient une propriété émergente** d'une suite de réactions du système, dans des cas d'interactions favorables. L'ensemble des évolutions possibles du système est déduit par le calcul, en se basant uniquement (sans autre paramètre) sur la connaissance de l'interaction.

Nous appliquons notre démarche à deux sous-systèmes interconnectés, dont l'objectif global est l'apprentissage d'actions réflexes.

Nous prouvons que le premier possède comme propriété émergente des facultés d'apprentissage par renforcement et d'apprentissage latent fiables et prédictibles.

Le deuxième, qui est ébauché, transforme un signal en une **information perceptive**. Il fonctionne par sélection d'hypothèses d'évolution du signal au cours du temps à partir d'une **mémoire**. Des contraintes internes à la mémoire déterminent les ensembles valides d'informations perceptives. Nous montrons, dans un cas simple, que **ces contraintes mènent à un équivalent du théorème de Shannon sur l'échantillonnage**.

Abstract

The long term goal of our work is the settlement of **reliable and predictable learning techniques** of basic behaviors in the robotics framework. This document is a starting point for this project.

As a first step, we argue that classical learning methods do not fulfill our request about reliability and predictibility. We think **the key point of this issue is the way the communication between the learning system and its environment is modelled**. We illustrate this point of view by giving a reinforcement learning example.

We introduce a **formalized** framework in which communication is seen as an **interaction**, as in physics. Two kinds of forces are applied to the system: the reaction of the system

is deduced, knowing the action of its environment and the fulfilment of **a set of internal constraints**. **Learning ability becomes an emerging property of the system** which is the result of several reactions over time. All the possible evolutions of the system are deduced from the prior knowledge about the interaction (with no need of other parameters).

We apply our technique to a set of two interconnected sub-systems, which global goal is to learn basic behaviors.

We prove that the first sub-system may possess as an emerging property (within some restrictive conditions) the abilities of reliable and predictable reinforcement learning and latent learning.

The second one is at a starting point. Its aim is to convert physical signals into what we call **perceptive information**. A selective process is involved in this task, in order to choose valid hypothesis about the evolution of the signal, from a set of hypothesis called **memory**. Internal constraints applied to the structure of the memory limit the valid sets of perceptive informations. In a simple case of memory, we show that these constraints lead to an **equivalent of the Shannon's sampling theorem**.

Remerciements

Je ne saurais débuter ce document sans évoquer les personnes qui m'ont toutes permis, d'une manière ou d'une autre, d'accomplir le travail qui s'achève avec la production de ce mémoire de thèse. Car personne ne saurait venir à bout d'une réalisation aussi conséquente qu'une thèse, s'étalant sur une durée aussi longue, sans l'aide directe ou l'accompagnement moral d'un nombre important de collègues, d'amis, de membres de la famille, voire de muses. En effet, la production d'un tel effort, que j'espère être de qualité, est le reflet immédiat des plaisirs, des joies et des passions qui m'ont poussé, coûte que coûte, à parvenir à la fin de mon travail : elle n'est possible que collectivement.

Mon entreprise a été excitante, mais aussi particulièrement périlleuse. J'estime avoir eu une chance exceptionnelle d'avoir été encadré par Claude Barret, qui a toujours su à la fois me laisser une très grande liberté d'action, mais aussi me conseiller efficacement au moment où il le fallait, tout en assumant le risque qu'il prenait avec moi, sur ce sujet de thèse. Je lui en témoigne une grande gratitude et un profond respect. Cette liberté d'action, mais aussi le sens des responsabilités qui en découle, est une constante du laboratoire LSC - CEMIF, qui montre la confiance que nos supérieurs hiérarchiques placent en nous, doctorants. Ce climat favorable à des relations humaines saines est dû, en grande partie, à Florent Chavand, directeur de ce laboratoire. Je le remercie pour cela, mais également pour le travail de relecture considérable qu'il a fourni, conjointement à Philippe Gaussier, Pierre-Yves Glorennec et Jeanny Herault, pour rendre mon document lisible (autant que possible) et, donc, me permettre de soutenir ma thèse dans de bonnes conditions. Je suis redevable à l'ensemble des membres du jury d'avoir fourni cet effort pour m'aider à progresser. Merci également à Jacques Droulez pour les conseils qu'il m'a donnés pour la troisième partie de ce document. Je remercie l'ensemble des personnes qui m'ont permis d'améliorer mes connaissances à propos de certains domaines scientifiques auxquels j'étais peu familier. Un grand merci à Vincent Vigneron pour l'ensemble des remarques pertinentes qu'il m'a adressées à propos de problèmes statistiques (mais aussi pour certains moments inoubliables de franche rigolade), mais aussi à Luc Jaulin, qui m'a initié à la problématique du calcul sur les intervalles : ils se sont montrés tous les deux très disponibles pour répondre à mes questions. Un merci tout particulier à Denis, passionné par les maths (et par plein d'autres choses autrement plus concrètes) et passionnant : il m'a été d'une grande aide, scientifique, morale, mais aussi culinaire.

Les journées de travail ont souvent été longues, mais jamais ennuyeuses. Je ne peux que remercier mes collègues et amis du laboratoire LSC - CEMIF. J'ai une pensée toute particulière pour Samir, toujours de bonne humeur et disponible pour aider, avec qui j'ai partagé beaucoup de choses : entre autres, les galères du samedi, lorsque les machines sont plantées, qu'il n'y a plus d'encre dans l'imprimante, et qu'il faut rendre un article pour le lundi matin ; les congrès (Lausanne, *where he saved my life*, ou Bourges) ; les grecs (très important, voire capital) ; mais surtout beaucoup de bons moments. Je pense également à Nicolas, dit « monsieur l'intendant » pour avoir réussi l'exploit de tenir le bar du laboratoire pendant plus d'une année (c'est-à-dire une éternité !), tout en pensant à l'ami Fred qui aime bien boire une bière de temps en temps. Merci également à Hamid, Naima, Yves, Djamel, Hichem, Salim, Omar et à l'ensemble des autres doctorants ; j'ai apprécié (et j'apprécie toujours) les pâtisseries orientales. Enfin, j'ai une pensée pour les anciens doctorants, qui ont été vers d'autres cieux : mon ami Mudar et Hafid également.

Cette thèse est dédiée à mes proches, qui m'ont encouragé sans contrepartie et qui ont dû

subir mes sautes d'humeur quand cela n'allait pas. C'est enfin achevé.

Notre âme n'aurait pas sujet de vouloir demeurer jointe à son corps si elle ne pouvait ressentir les passions

René Descartes

Ce n'est pas parce que les choses sont difficiles que nous n'osons pas les entreprendre, c'est parce que nous n'osons pas les entreprendre qu'elles semblent difficiles.

Sénèque

Table des matières

Avant-propos

1 Mise en situation xvii
2 Thèse xviii
2.1 Postulat xviii
2.2 Mise à l'écart des modélisations classiques xviii
3 Éléments de modélisation xix
3.1 Causes possibles du non respect de notre postulat par les modélisations classiques xix
3.2 Démarche associée à notre modélisation xx
3.3 Modèle du système apprenant xxii
3.4 Principaux résultats xxiv
3.5 Limitations xxiv
4 Plan du document xxiv

Partie I Apprentissage d'objectif (AO) **1**

Chapitre 1
Influence du contexte sur l'apprentissage par renforcement

1.1 Introduction 3
1.1.1 Idées directrices 3
1.1.2 Principaux résultats 4
1.1.3 Guide du chapitre 4
1.2 Cadre général de l'apprentissage par renforcement (AR) 5
1.2.1 Le mécanisme d'AR schématisé - les termes clés 5
1.2.2 Références générales 6
1.2.3 Positionnement de l'AR par rapport à d'autres méthodes d'apprentissage 6
1.2.4 Problématique et méthodes de résolution liées à l'AR 7
1.2.5 Conditions de convergence des algorithmes d'AR 8

1.2.6 Causes de difficultés dans l'utilisation des algorithmes d'AR 9
1.2.7 Lien entre incertitude, imprécision, fiabilité et prédictibilité 10
1.3 Outils d'étude de l'incertitude due au contexte de l'AR 11
1.3.1 Introduction 11
1.3.2 Qu'entendons-nous par « qualité » du contexte d'apprentissage? . . . 11
1.3.3 Notations 12
1.3.4 Contexte idéal et quasi-idéal - Propriété (P_ϵ) 12
1.3.5 Exemples de contextes vérifiant ou ne vérifiant pas (P_ϵ) 13
1.3.6 Information associée à l'exécution d'une action 13
1.3.7 Mesures utilisant l'entropie de Shannon 14
1.3.8 Protocole de calcul des mesures H_1 et H_2 16
1.3.9 Modélisation du flux d'erreurs dû au contexte d'apprentissage 16
1.3.10 Modélisation d'un flux d'erreurs mono-causal 18
1.3.11 Modélisation d'un flux d'erreurs bi-causal dépendant de l'état initial du système 20
1.4 Expérimentations autour du problème du pendule inversé 21
1.4.1 Objectif 21
1.4.2 Analyse préliminaire des résultats antérieurs sur l'influence du bruit de mesure sur l'apprentissage par AR 22
1.4.3 Protocole expérimental 24
1.4.4 Analyse des mesures H_1 et H_2 lorsque la qualité des données d'entrée est dégradée 26
1.4.5 Découverte des sources d'erreur du contexte de l'apprentissage 28
1.4.6 Relation entre H_1 et ϵ_2 33
1.4.7 Conclusion 34

Chapitre 2
Algorithme d'AO défini dans un contexte idéal

2.1 Introduction 39
2.1.1 Idée directrice du chapitre 39
2.1.2 Principaux résultats 40
2.1.3 Plan du chapitre 40
2.2 Modélisation du sous-système d'apprentissage d'objectif 40
2.2.1 Méthodologie 40
2.2.2 Spécification du sous-système 41
2.2.3 Contraintes appliquées au sous-système 42
2.2.4 Action de l'environnement et réaction du sous-système 43

2.3 Algorithme d'apprentissage Constraint based Learning (CbL) 43
2.3.1 Introduction 43
2.3.2 Exemple de propagation des marquages du graphe d'état 43
2.3.3 Algorithme CbL 44
2.3.4 Remarques 44
2.4 Résultats théoriques concernant l'algorithme CbL(1) 47
2.4.1 Problèmes théoriques 47
2.4.2 Convergence de la phase de propagation donnée par l'algorithme 2.1 . 48
2.4.3 Problème d'unicité des valeurs des marquages obtenues grâce à l'algorithme de propagation 49
2.4.4 Signification de la valeur des marquages 53
2.4.5 Exploration de l'espace d'états 54
2.4.6 Convergence et prédictibilité de l'algorithme CbL 55
2.4.7 Utilisation de l'algorithme CbL(α) 56
2.5 Propriété d'incrémentalité de l'algorithme CbL 56
2.5.1 Qu'entendons-nous par « incrémentalité » ? 56
2.5.2 Découverte de la suppression d'une cible 56
2.5.3 Découverte de l'ajout d'une cible 57
2.5.4 Découverte de la suppression d'un obstacle 57
2.5.5 Découverte de l'ajout d'un obstacle 58
2.5.6 Conclusion : lien entre la capacité d'incrémentalité de l'algorithme CbL et l'invariant structurel engendré par (P_ϵ) 58
2.6 Problème du labyrinthe 58
2.6.1 Introduction 58
2.6.2 Position du problème 59
2.6.3 Protocole d'apprentissage 59
2.6.4 Résultats 61
2.7 Problème de navigation d'un robot mobile 69
2.7.1 Introduction 69
2.7.2 Position du problème 69
2.7.3 Préparation du contexte d'apprentissage pour le problème de navigation du robot Khepera 70
2.7.4 Protocole expérimental 76
2.7.5 Résultats 76
2.8 Conclusion 80

Annexe A
Éléments relatifs au chapitre 1

A.1 Techniques d'apprentissage par renforcement 83
A.1.1 Avertissement - remerciements 83
A.1.2 Architecture et algorithme Q-Learning 83
A.2 Expériences autour du pendule inversé 85
A.2.1 Le problème du pendule inversé 85
A.2.2 Valeur des paramètres internes choisis pour l'algorithme Q-Learning . 85
A.3 Éléments de calcul de probabilité 86
A.4 Calcul d'un estimateur $\hat{\epsilon}$ par la méthode du maximum de vraisemblance . . . 87

Partie II Apprentissage perceptif (AP) 89

Chapitre 3
Modèle paramétrique du sous-système d'AP 91

3.1 Introduction 91
3.1.1 Idées directrices 91
3.1.2 Plan du chapitre 92
3.2 Sous-système d'apprentissage perceptif 92
3.2.1 Caractéristiques de l'information perceptive 92
3.2.2 Modèle global 93
3.2.3 Modèle d'un événement élémentaire 94
3.2.4 Mécanisme de sélection des hypothèses valides 94
3.3 Mécanisme de sélection pour un ensemble fini d'hypothèses 95
3.3.1 Introduction 95
3.3.2 Algorithme 96
3.3.3 Précisions concernant l'algorithme de sélection 96
3.4 Mécanisme de sélection pour un ensemble infini d'hypothèses 98
3.4.1 Introduction 98
3.4.2 Constitution de la mémoire - notations 98
3.4.3 Formalisation de l'ensemble S(t) des solutions - résolution d'un problème d'inversion ensembliste 100
3.4.4 Méthode de résolution 100
3.4.5 Algorithme 102
3.5 Contraintes appliquées à la mémoire 102
3.5.1 Introduction 102

3.5.2 Notion d'événement rare . . . 102
3.5.3 Le problème « D » . . . 104
3.5.4 Extensions du problème « D » . . . 106
3.5.5 Contrainte d'observabilité CO . . . 107
3.5.6 Contrainte d'unicité (CU) . . . 108
3.6 Caractère générique de notre modélisation - Lien avec des modélisations paramétriques existantes . . . 109
3.6.1 Prédiction : liens avec le filtrage de Kalman . . . 110
3.6.2 Possibilité de bifurcation . . . 111
3.6.3 Possibilité d'avoir des hypothèses possédant des valeurs de h différentes : lien avec les approches multi-résolutions . . . 113
3.6.4 Possibilité de valider simultanément plusieurs hypothèses : lien avec la problématique de la séparation de sources . . . 113
3.7 Conclusion . . . 114
3.7.1 Résultats obtenus . . . 114
3.7.2 Travaux à effectuer . . . 115

Chapitre 4
Résultats théoriques et perspectives à propos de l'information perceptive

4.1 Guide du chapitre . . . 117
4.2 Résolution de CO dans le cas d'une mémoire possédant une hypothèse . . . 117
4.2.1 Introduction . . . 117
4.2.2 Notations - Formulation des deux contraintes de CO . . . 118
4.2.3 Théorème d'existence d'une mémoire respectant CO et CU . . . 120
4.2.4 Comparaison informelle entre CO et la contrainte imposée par le théorème d'échantillonnage de Shannon . . . 123
4.2.5 Limitations des mémoires à une hypothèse - Extension du résultat d'existence à une catégorie d'ensembles finis d'hypothèses . . . 126
4.2.6 Conclusion . . . 127
4.3 Conclusion générale des deux premières parties de notre document . . . 127
4.3.1 Méthodologie . . . 127
4.3.2 Résultats théoriques . . . 128
4.3.3 Algorithmes de sélection . . . 128
4.4 Perspectives . . . 129
4.4.1 Introduction . . . 129
4.4.2 Conjecture à propos des mémoires possédant un ensemble infini d'hypothèses . . . 129

4.4.3 Piste de recherche sur l'AP 131
4.4.4 Généralisation à l'utilisation de plusieurs signaux d'entrée 133
4.4.5 Genèse de l'information perceptive à l'aide d'actions réflexes 133

Annexe B
Éléments relatifs au chapitre 4

B.1 Exemple d'information perceptive pour un signal mono-dimensionnel 135
B.1.1 Introduction et notations 135
B.1.2 Fiabilité des informations perceptives obtenues 137
B.1.3 Probabilité de découverte au hasard d'un segment orienté 144
B.1.4 Découverte de n tendances consécutives pour un signal de densité de probabilité uniforme 145
B.2 Relation entre le paramètre ϵ et le postulat de rareté de l'information perceptive 147
B.3 Preuves concernant la fiabilité de la détection de l'information perceptive . . 149
B.3.1 Introduction - Notations 149
B.3.2 Fiabilité de la détection d'une information perceptive 149
B.3.3 Preuve de la proposition 8 152
B.3.4 Preuve du théorème 1 (paragraphe 4.2.3, page 120) 153

Partie III A parte : Réflexion informelle autour de notre modélisation 161

Chapitre 5
Plausibilité de notre modèle au regard du vivant

5.1 Idées directrices 163
5.2 Caractéristiques de l'apprentissage perceptif 164
5.2.1 Différenciation et unification 164
5.2.2 Influence du contexte sur le mécanisme de différenciation 165
5.2.3 Utilisation de contraintes 166
5.3 Caractéristiques de l'information perceptive 167
5.3.1 Notion d'information perceptive 167
5.3.2 Postulat d'existence de l'information perceptive 168
5.3.3 L'information perceptive est soumise au paradoxe de l'évidence 168
5.3.4 Rôle des signaux perceptifs internes 170
5.3.5 Généralisation de la notion de perception aux signaux internes accompagnant le mouvement 171
5.3.6 Mécanisme d'anticipation 172

5.4 Importance des notions de fiabilité et de prédictibilité 175
5.4.1 Introduction 175
5.4.2 Nature de la représentation mentale : cohérence entre l'anticipation et le fait observé 175
5.4.3 Sensation de sécurité comme moteur de l'apprentissage 176
5.4.4 L'optimisation vue comme une capacité à générer un grand nombre de catégories perceptives 178

Chapitre 6
Positionnement de notre démarche scientifique

6.1 Introduction 181
6.2 Différentes approches de l'Intelligence 182
6.2.1 Introduction 182
6.2.2 Idées fondatrices associées à la machine de Turing - hypothèse béhavioriste 182
6.2.3 Idées associées à l'approche cognitive de l'intelligence 186
6.2.4 Idées associées à l'approche biologique de l'intelligence 190
6.2.5 Réflexion à propos de la démarche fonctionnaliste 192
6.2.6 Conclusion - Liens avec la biologie et avec le concept de la machine de Turing 197
6.3 Une certaine notion de la réalité 198
6.3.1 Introduction 198
6.3.2 Lien entre information perceptive et observation 198
6.3.3 Notion de la réalité, dérivée de l'information perceptive 199

Avant-propos

1 Mise en situation

Notre travail s'est construit autour d'une réflexion originale, dont **le cadre applicatif est l'apprentissage d'actions réflexes d'un robot mobile** . Beaucoup de recherches ont déjà été effectuées sur ce sujet, en particulier à partir des travaux de Brooks [Brooks, 1986] ou de Arkin [Arkin, 1989], donnant lieu à des applications fonctionnelles [Mataric, 1992]. Cependant, nous devons constater que, même si des résultats prometteurs sont d'ores et déjà obtenus, **il existe un fossé entre les travaux théoriques à propos des algorithmes d'apprentissage et l'élaboration de solutions fonctionnelles** ; par exemple, Pendrith a bien souligné cela dans le cadre de l'apprentissage par renforcement [1] en robotique mobile [Pendrith et McGarity, 1998].

Il semble même que des techniques développées spécialement pour la robotique mobile, ne possédant que peu de fondements théoriques, donnent des résultats équivalents sinon meilleurs à ceux obtenus par des méthodes théoriquement fondées. Parmi les premières, on peut signaler celles qui sont **inspirées par l'étude du vivant** (sciences cognitives ou neurosciences) : le développement des études sur les *animats* va dans ce sens ([Meyer et Wilson, 1991],[Gaussier et Zrehen, 1995]).

Ce constat a constitué le point de départ de notre travail de thèse et a, en partie, motivé notre volonté de constituer un **modèle biologiquement plausible** (chapitre 5). D'autre part, il soulève deux questions fondamentales complémentaires :

- pourquoi l'emploi de techniques formalisées utilisées actuellement pose-t-il des difficultés en pratique ?
- à l'inverse, quels principes non explicitement formalisés les méthodes inspirées du vivant utilisent-elles, qui permettent d'obtenir un résultat fonctionnel et robuste ?

Notre travail à long terme a pour ambition de répondre à ces deux questions, qui nous semblent fondamentales, tout en proposant un formalisme adapté au cadre applicatif de l'apprentissage en robotique mobile. Ainsi, il faut voir ce document comme l'ébauche d'une réponse possible, qui doit être complétée par des travaux ultérieurs.

Cet avant-propos introduit la thèse que nous défendons, précise les termes-clés que nous utiliserons dans la suite et indique notre plan.

1. Le lecteur pourra se reporter aux articles de référence de Barto et Sutton [Barto et al., 1983], puis de Watkins [Watkins, 1989]

2 Thèse

2.1 Postulat

Avant tout, il est important de rappeler que le choix de la modélisation d'un problème implique implicitement ce qui va être possible de résoudre, mais induit aussi des **limitations intrinsèques**. Nous pensons que, pour répondre aux deux questions posées en préambule, il est important de savoir dans quelle mesure ces limitations **théoriques**, inévitables quelle que soit la modélisation choisie, sont **compatibles** avec le problème qui doit être résolu **en pratique**. Notre thèse part du postulat suivant :

Postulat 1 *Il existe une modélisation d'un apprentissage d'actions réflexes par un robot mobile, qui répond à trois exigences :*

1. *on ne possède pas de connaissance a priori sur l'environnement ni sur le fonctionnement des capteurs ou des effecteurs du robot*[1]
2. *les limitations intrinsèques de la modélisation sont compatibles avec un environnement réel non contraint*
3. *la modélisation est formalisée et le ou les algorithmes qui en découlent possèdent des preuves de convergence, ainsi que des caractères de* **fiabilité** *et de* **prédictibilité**

Dans la suite du document, l'acception de la **fiabilité** sera identique à celle qui est utilisée en sûreté de fonctionnement, c'est-à-dire qu'elle désigne la propension d'un système à éviter des événements dommageables (le robot se cogne contre un mur, par exemple), calculée **avant l'expérience** en termes de fréquences d'apparition. D'autre part, l'acception de la **prédictibilité** sera la propension à déterminer **avant l'expérience** l'ensemble des configurations[2] possibles que le système apprenant peut adopter au cours de l'expérimentation[3].

2.2 Mise à l'écart des modélisations classiques

Dans le cas de la robotique mobile, la littérature montre plusieurs types généraux de modélisation.

Suivant la précision des connaissances qu'on possède *a priori* sur l'environnement du robot et le comportement de ses capteurs, deux classes de possibilités s'offrent au concepteur[4] :

- utiliser des méthodes inspirées de l'automatique
- employer des heuristiques ou des algorithmes d'apprentissage

La première approche est, du moins pour le « puriste », conditionnée par deux hypothèses fortes :

- modéliser le robot, son environnement et leur interaction, avec une précision suffisante
- pouvoir montrer la stabilité de la loi de commande du robot associée à ce modèle

Lorsque ces deux conditions sont remplies, le concepteur obtient la garantie que son système est fonctionnel et fiable, selon notre acception donnée précédemment. Cependant, en pratique,

1. Cela signifie qu'il peut être possible de rajouter de la connaissance *a priori*, mais que cela ne doit pas être indispensable. 2. Nous raisonnerons par la suite en termes d'événements. Une configuration sera un ou des événements détectés par le système apprenant. 3. Nous sommes ici en dehors de toute interprétation en termes de qualité des configurations (favorables ou défavorables). 4. Bien entendu, dans un problème concret, ces deux catégories peuvent être utilisées conjointement.

les applications de robotique mobile remplissent rarement ces deux hypothèses. On peut néanmoins enfreindre une de celles-ci, mais on perd alors la garantie de stabilité de la loi de commande.

Ainsi, en pratique, **les conditions 1 et 2 relatives à notre postulat sont mises en défaut par la modélisation de l'automaticien**, car, si on cherche à prouver le bon fonctionnement du système dès sa conception, on est obligé de **contraindre l'environnement du robot et de maîtriser la qualité de ses capteurs**.

C'est pour éviter une démarche de preuve astreignante et rarement possible dans des environnements complexes ou inconnus que le choix d'une heuristique ou d'un algorithme d'apprentissage peut être intéressant. Parmi les techniques d'apprentissage **possédant des preuves de convergence**, et pouvant remplir les exigences de notre postulat, l'apprentissage par renforcement (AR) offre l'avantage de ne pas avoir besoin d'un modèle de l'environnement ; l'apprentissage s'effectue par essais/erreurs successifs, grâce à un retour d'information qui peut être très pauvre (binaire ou ternaire, en général).

Cependant, on constate en pratique que l'hypothèse markovienne, permettant de prouver la convergence de l'AR, est rarement respectée dans le cadre applicatif de la robotique mobile. D'autre part, certains paramètres très influents sur le résultat de l'apprentissage doivent être fixés arbitrairement avant le début de l'apprentissage (c'est le cas du paramètre de température, qui règle le dilemme exploration/exploitation). Enfin, l'AR suppose qu'il existe déjà un moyen de construire les états du système. Nous reviendrons plus précisément sur l'AR au cours du chapitre 1. Nous y argumentons, à partir d'un exemple très simple, que l'ensemble de ces facteurs, que nous appelons **contexte d'apprentissage**, ne permet pas de garantir les propriétés de fiabilité du résultat d'un apprentissage et de prédictibilité de l'algorithme d'AR, ce qui n'est pas compatible avec la troisième condition de notre postulat.

Nous sommes donc amenés à rejeter les deux archétypes principaux de modélisation, qui ne respectent pas notre postulat.

3 Éléments de modélisation

3.1 Causes possibles du non respect de notre postulat par les modélisations classiques

Une modélisation est associée à des postulats qui prennent racines dans l'histoire des sciences. Il nous a semblé important de citer certains d'entre-eux, afin de donner des arguments en faveur de la mise à l'écart des modélisations classiques dans le cadre de notre postulat : cela est développé dans le chapitre 6, dans la section 6.2. Toutefois, nous ne prétendons pas être exhaustifs, ni même formuler un débat critique autour de ce thème.

Donnons dès à présent les pistes de cette réflexion, qui nous permettrons de créer notre propre modélisation. L'idée essentielle est que les deux archétypes que nous avons décrits sont, d'une certaine manière, trop exigeants et durcissent au moins une des conditions associées à notre postulat. **Nous pensons que ces exigences peuvent créer l'incompatibilité avec nos postulats** : nous cherchons donc un compromis pragmatique.

Concernant la modélisation de l'automaticien, l'obtention d'une **preuve de stabilité de la loi de commande** implique, si le modèle coïncide avec la réalité, la *prédictibilité* de l'algorithme

de commande et la *fiabilité* de celui-ci ce qui satisfait la troisième condition de notre postulat. Mais elle a également d'autres conséquences :

1. il n'existe théoriquement aucun cas pour que la loi de commande devienne instable : l'ensemble des configurations possibles du système sont favorables
2. le système possède un ou des états d'équilibre vers lesquels l'état du système converge

Concernant la modélisation d'un système d'apprentissage comme celui de l'AR, il nous semble que **la volonté d'obtention d'une solution optimale**[1] crée une condition à la fois non indispensable et peu réaliste (dans des cas réels, on n'arrive pas à une solution optimale, mais sous-optimale dans le meilleur des cas). L'exigence d'optimalité impose, en théorie, une convergence de l'algorithme en temps infini [Dayan et Sejnowski, 1994] et l'existence d'un contexte d'apprentissage construit autour de l'algorithme qui influence le résultat de l'apprentissage et restreint l'adaptabilité du système apprenant.

D'une manière générale, la démarche de l'automaticien est compatible avec celle que nous souhaitons employer, mais elle est déterministe ce qui la rend trop rigide, alors que l'autre catégorie de démarche s'appuie sur la notion de système stochastique, qui est trop souple car celle-ci ne permet pas de contrôler **avant l'expérience** la probabilité d'apparition d'événements, qui est à la base de notre définition de la fiabilité.

Enfin, on constate que **chacune des deux démarches est orientée de manière à trouver les configurations favorables du système**. Dans la modélisation de l'automaticien, cela se traduit par l'impossibilité d'établir une preuve de stabilité autorisant une proportion, même très faible, de configurations défavorables ; pourtant, cela nous suffirait. D'autre part, dans la modélisation d'un système apprenant, cela diminue également le domaine de validité de l'algorithme, pour lequel on a la preuve que celui-ci converge.

Paradoxalement, en suivant notre dernière remarque, nous pensons que si on souhaite trouver une modélisation pour laquelle on est certain de la convergence de l'algorithme, **il faut abandonner l'idée que celui-ci est construit pour apprendre** (c'est-à-dire pour rechercher uniquement les conditions favorables du système). Dans la suite de ce document, c'est cela qui nous amènera à considérer **l'apprentissage comme une faculté émergente d'un système placé dans des conditions favorables**. Le terme d'émergence est choisi dans ce cas, car il signifie que l'apprentissage n'est pas une propriété intrinsèque du système, mais un phénomène observable dans certaines situations uniquement.

3.2 Démarche associée à notre modélisation

Une modélisation comporte une démarche et un ou plusieurs modèles sur lesquels on va appliquer cette démarche. Celle que nous avons choisi d'adopter - et dont nous donnons un exemple détaillé dans le chapitre 2 - repose sur le **concept de viabilité**[2], qui va remplacer celui d'optimisation par rapport à un objectif précis (ou une configuration favorable du système).

À l'image d'un système physique soumis à des forces, nous faisons l'hypothèse que le système apprenant est un système ouvert, en interaction avec son environnement et **soumis à un ensemble de contraintes internes, qui doivent être obligatoirement respectées à tout moment**. Ces contraintes sont une généralisation de systèmes d'équations, qui permet de lier

1. L'AR est une méthode d'optimisation dérivée de la Programmation Dynamique [Bellman, 1957]. 2. On pourra se référer à [Aubin, 1991]

implicitement les différents paramètres internes du modèle, d'une manière moins rigide qu'un système d'équations. Nous verrons au cours des chapitres 3 et 4 que l'application de certaines contraintes peut aboutir à la résolution, à chaque pas de temps, d'**un problème inverse ensembliste**[1], dont le résultat est un ensemble de solutions satisfaisant les contraintes. Dans le chapitre 2, les contraintes reviendront simplement à spécifier un système d'équations.

Voici comment, dans ce cadre, nous aboutissons à la notion d'apprentissage. L'**action** de l'environnement a tendance à rompre les contraintes, qui doivent **toujours** pouvoir être rétablies par **réaction** en modifiant certaines variables associées au modèle du système apprenant. L'ensemble des réactions du système entraîne une modification de celui-ci, qui peut être interprétée, dans certains cas, comme un apprentissage : cela constitue les configurations favorables du système.

L'étude antérieure à l'expérience se compose de deux phases. La première consiste à spécifier les points suivants :

1. un modèle du système
2. les contraintes qui lui sont appliquées
3. la manière dont l'environnement agit sur le système
4. la manière dont le système réagit pour rétablir ses contraintes internes

Dans la seconde, on étudie l'ensemble des évolutions possibles du système, qui aboutissent à des configurations favorables mais aussi défavorables.

Enfin, à partir de cette étude préalable, on peut déterminer l'ensemble des environnements pour lesquelles le système évolue favorablement (c'est-à-dire qu'il peut apprendre). Les caractéristiques de cet ensemble permettent d'affirmer s'il est compatible avec ce à quoi le robot aura à faire face en pratique.

Dans notre cas, la démarche de preuve s'effectue en deux étapes, dont un exemple applicatif complet est donné dans la section 2.4 :

1. montrer qu'il est toujours possible de détecter une rupture des contraintes, puis de rétablir ces contraintes, **pour tout environnement**
2. établir l'ensemble des évolutions possibles du système (favorables et défavorables) et caractériser le sous-ensemble des évolutions favorables

Cette dernière étape permet de déterminer *a posteriori* les hypothèses restrictives concernant les environnements « favorables ».

En conclusion, la notion de fiabilité (troisième condition du postulat) est liée à la maîtrise de la probabilité que le système reste dans une configuration favorable, alors que la prédictibilité est assurée si on sait établir, avant l'expérience, l'ensemble des évolutions possibles du système.

Si l'application de notre démarche à un modèle permet d'extraire des hypothèses assez larges pour être compatibles avec les environnements réels, la deuxième condition du postulat est alors vérifiée.

1. On utilisera ici des éléments de calcul sur les intervalles. Le lecteur pourra se reporter à [Moore, 1979]

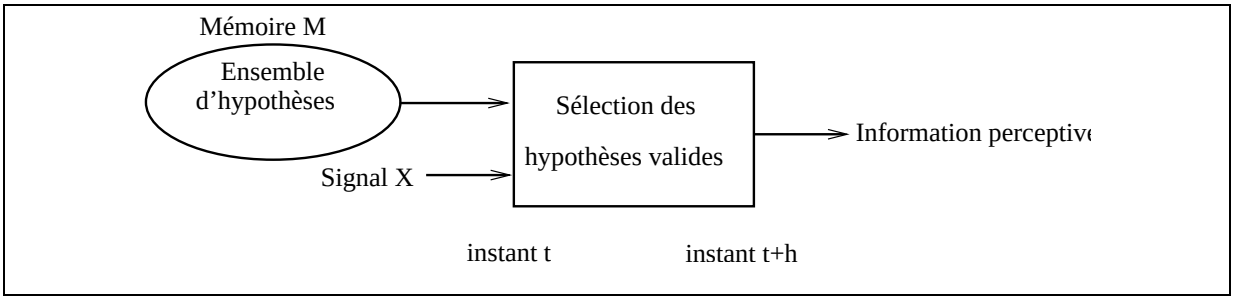


Fig. 1 – Dépendance de l'information perceptive avec la mémoire du système apprenant

3.3 Modèle du système apprenant

Nous souhaitons trouver un modèle d'un système qui apprend des actions réflexes, c'est-à-dire des fonctionnalités de bas niveau. Dans la littérature, il existe de nombreux modèles. On peut citer celui de Brooks [Brooks, 1986], celui de Arkin [Arkin, 1989], celui de Rasmussen [Rasmussen, 1986] ou celui de Gaussier [Gaussier et Zrehen, 1995].

L'ensemble de ces modèles considère que **l'interaction entre le système et son environnement se résume à un échange d'informations**. Ainsi, ceux-ci supposent que l'existence et l'objectivité de l'information sont garanties *a priori*.

L'objectivité signifie qu'on suppose que ces informations existent et ont un sens **indépendamment du système apprenant qui les reçoit** avec plus ou moins de précision. À l'entrée du système, se situe donc un sous-système de traitement du signal, utilisant les règles d'échantillonnage et de quantification de Shannon [Shannon, 1948].

L'interprétation de l'information faite par Shannon induit l'existence de la notion de bruit de mesure (différence entre le signal parfait, dont on suppose l'existence, et le signal réel). **Nous avons l'intuition, sans pouvoir le montrer formellement dans ce document, que cette interprétation de l'information n'est pas compatible avec l'exigence de maîtrise** *a priori* **de la fiabilité du système**. Cela se traduit, dans les faits, par l'existence d'un contexte d'apprentissage dont nous montrons dans le chapitre 1 qu'il peut être incompatible avec notre notion de la fiabilité.

Nous opposons l'objectivité forte de l'information de Shannon avec l'objectivité faible[1] de la notion d'information perceptive, que nous introduisons dans ce document (voir les chapitres 3 et 4). L'objectivité faible suppose que **l'information perceptive est dépendante du système apprenant** : la figure 1 illustre le modèle de dépendance que nous avons choisi. La mémoire, qui fait partie du système apprenant, est **un ensemble d'événements concernant l'évolution future de l'environnement**, qui peuvent **potentiellement** se produire. L'information perceptive est le résultat d'un **processus dynamique de sélection** à partir de cet ensemble.

La mémoire faisant partie de notre système, elle est soumise à des contraintes, appelées **Contrainte d'Observabilité (CO)** et **Contrainte d'Unicité (CU)**, qui vont guider l'évolution de la mémoire au cours du temps. L'idée qui est à la base de ces contraintes est que **la détection d'une information perceptive est, théoriquement, un événement rare** (si on

1. Les termes d'*objectivité forte* et d'*objectivité faible* sont empruntés à d'Espagnat [d'Espagnat, 1994].

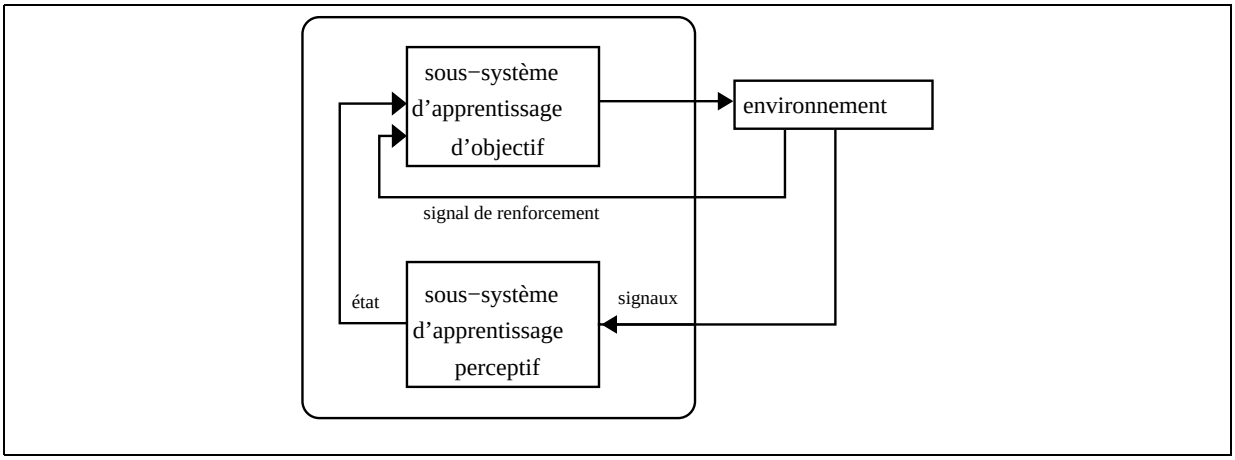


Fig. 2 – Modèle de notre système apprenant

considère l'ensemble des environnements possibles), mais doit être, **en pratique, un événement quasi-systématique**. Ce paradoxe apparent, décrit sous le nom de **problème D** (voir la section 3.5), est fondé sur l'hypothèse, non restrictive en pratique, que l'ensemble des environnements auxquels le système fait face en réalité est beaucoup plus petit que l'ensemble des environnements possibles.

Comme notre démarche nous y invite, la modification de la mémoire devra pouvoir être interprétée comme un apprentissage, que nous nommons **Apprentissage Perceptif** (AP) et dont nous venons de donner l'hypothèse théoriquement restrictive.

Il est intéressant de constater que l'information perceptive, bien qu'étant différente de l'information de Shannon, possède une propriété similaire : dans un cas particulier, nous avons montré un **équivalent du théorème d'échantillonnage** (théorème 1, chapitre 4).

Outre l'information perceptive, le modèle de système apprenant intègre également un apprentissage latent [1] et un apprentissage par conditionnement. Dans ce document, nous intégrons ces deux apprentissages dans le terme **Apprentissage d'Objectif** (AO). L'apprentissage latent va aboutir à un **modèle interne de l'environnement**, qui est constitué d'un graphe d'états dont les sommets sont des informations perceptives (que nous nommerons également *états*) et dont les arcs sont les actions possibles du système. Le conditionnement s'établit en supposant l'existence d'états terminaux positif et négatif, possédant un marquage (ou une Q-value) déterminé.

Nous appliquons un ensemble de contraintes sur les Q-values associées à chacun des sommets du graphe, qui va engendrer la propagation des valeurs des états terminaux (voir la section 2.2). L'algorithme qui découle de ces contraintes est appelé **CbL** pour *Constraint based Learning.*

En résumé, notre modèle de système apprenant comporte deux blocs, l'un soumis à l'AO (sous-système d'apprentissage d'objectif) et l'autre soumis à l'AP (sous-système d'apprentissage perceptif). Un schéma bloc est donné par la figure 2.

1. L'apprentissage latent désigne, à l'origine, l'apprentissage d'un environnement par un rat, sans avoir reçu le moindre renforcement positif (nourriture ou boisson). Ce phénomène a été mis en évidence par Seward [Seward, 1949] et a été utilisé dans des algorithmes comme l'ACS de Stolzmann [Stolzmann, 1998].

3.4 Principaux résultats

Nous avons étudié les deux sous-systèmes séparément.

Pour le sous-système d'apprentissage d'objectif, la démarche associée à notre modélisation est totalement achevée. Nous donnons les résultats théoriques concernant l'algorithme CbL (rétablissement des contraintes, convergence et cas favorables pour lesquels on peut interpréter l'évolution du système comme un apprentissage). L'algorithme CbL est appliqué à un problème de labyrinthe et à la navigation d'un robot mobile simulé. Nous constatons que l'utilisation d'un apprentissage latent permet d'accélérer considérablement la vitesse d'apprentissage par rapport à une méthode d'apprentissage par renforcement classique (dérivée du Q-Learning).

L'étude du sous-système d'apprentissage perceptif est ébauchée. Le résultat principal concerne l'existence d'un équivalent du théorème de Shannon sur l'échantillonnage.

3.5 Limitations

Dans l'idéal, il aurait été préférable de considérer le système apprenant **dans sa globalité**. De plus, cette séparation n'est pas biologiquement plausible, ce qui est un point limitant de notre modèle.

L'approche de l'apprentissage perceptif en est à l'état d'ébauche. La dynamique de l'évolution de la mémoire n'est pas formellement abordée ; **une solution est proposée en perspective**. Cette limitation est due à deux difficultés :

- dans le cas général, il est théoriquement difficile de savoir si une mémoire donnée respecte les contraintes d'observabilité et d'unicité
- en admettant que le point précédent est résolu, il faut pouvoir modifier la mémoire au cours du temps, de manière à être certain de respecter ces contraintes

Enfin, pour le moment, le formalisme est développé en utilisant des signaux mono-dimensionnels.

4 Plan du document

Ce document est composé de trois parties indépendantes. Le découpage est principalement fait par rapport au découpage en deux sous-systèmes, choisi pour notre modèle.

La première partie traite de l'AO. Le chapitre 1 donne des éléments de réponse au fait qu'une modélisation classique, en l'occurrence l'Apprentissage par Renforcement, ne permet pas de garantir les conditions de fiabilité et de prédictibilité données dans notre postulat. Nous illustrons cela par un exemple d'AR classique (le pendule inversé). En outre, nous associons les performances d'algorithmes d'apprentissage d'atteinte d'un objectif à une notion de la qualité du **contexte d'apprentissage** qui dépend, en particulier, de la manière dont l'expérimentateur gère la perception du système (association perception/état). Une caractérisation d'un type de contexte « favorable », *appelé contexte idéal*, est donnée. Dans ce contexte, un algorithme d'apprentissage par renforcement, appelé CbL (pour Constraint based Learning), est proposé. Nous montrons théoriquement qu'il satisfait nos exigences de fiabilité et de prédictibilité, dans ce contexte idéal, ce qui induit **une hypothèse limitative de fonctionnement de CbL**. Nous donnons deux exemples applicatifs : un problème de navigation d'un robot miniature de type

Khepera, ainsi qu'un problème de type « labyrinthe ». Cette étude est menée dans le chapitre 2.

La deuxième partie du document est consacrée à l'étude du sous-système d'apprentissage perceptif et des contraintes qui lui sont appliquées. L'AP sera abordé en guise de perspective de notre travail. Notre étude porte essentiellement sur des signaux mono-dimensionnels. Toutefois, nous proposons un fondement possible d'un mécanisme général fusionnant ces signaux. Le chapitre 3 décrit le sous-système d'apprentissage perceptif, introduit les objets s'intégrant à ce processus et donne un algorithme de sélection des hypothèses valides, qui n'utilise aucun paramètre propre. Il exploite une entrée appelée « mémoire », qui est constituée d'un ensemble d'hypothèses sur l'évolution future du signal (mécanisme d'anticipation). Cette mémoire possède des paramètres dont la valeur n'est pas précisée *a priori*: ceux-ci caractérisent en particulier la nature et la « forme » des hypothèses.

Le chapitre suivant se focalise sur cette mémoire. Il montre que l'ensemble des hypothèses doit respecter des contraintes (contrainte d'observabilité (CO), contrainte d'unicité (CU)), ce qui permet alors de donner une relation entre les paramètres des hypothèses. Savoir si un ensemble (fini ou infini) d'hypothèses respecte les contraintes (CO) et (CU) est un problème particulièrement difficile. Nous résolvons mathématiquement le cas d'une mémoire réduite à une hypothèse. Nous montrons dans ce cas que, si la mémoire est correctement construite, **la détection de l'information perceptive est fiable, même en présence d'un taux de données aberrantes important** ; le cas d'une fiabilité totale ne peut s'obtenir que si la durée d'observation est infinie. Nous retrouvons en cela la démarche de Shannon dans sa théorie de la transmission du signal. Nous montrons que, sous certaines conditions (indépendance des hypothèses constituant la mémoire), les calculs pour un ensemble fini se ramènent au cas unitaire. Enfin, nous donnons des éléments numériques de preuve pour un exemple particulier d'ensemble infini.

En guise de conclusion à ce chapitre et à cette partie, nous donnons des voies de recherche possibles, permettant d'établir le mécanisme de réaction de la mémoire, dont on suppose l'existence, qui pourrait faire émerger l'AP.

Enfin, une troisième partie présente les aspects les plus importants ayant contribué à l'élaboration de notre travail préliminaire. Elle présente une réflexion informelle que nous avons effectuée en guise de préalable au travail que nous présentons dans les deux premières parties de ce document. Nous y expliquons comment nous avons choisi notre voie de recherche. Les axes de réflexion sont la cohérence de notre approche au regard des sciences du vivant (chapitre 5), ainsi que le positionnement de notre démarche scientifique (chapitre 6).

Références

Arkin, R. (1989). Toward the unification of navigational planning and reactive control. In *AAAI Spring Symp. Robot Navigation Working Notes*, pages 1–5.

Aubin, J. (1991). *Viability theory (Systems and Controls)*. Springer-Verlag.

Barto, A., Sutton, R., and Anderson, C. (1983). Neurolike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC13 :834–846.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press, Princeton, NJ.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1) :14–23.

Dayan, P. and Sejnowski, T. (1994). Td(λ) converges with probability 1. *Machine Learning*, 14 :295–301.

d'Espagnat, B. (1994). *Le réel voilé*. Fayard.

Gaussier, P. and Zrehen, S. (1995). Perac : A neural architecture to control artificial animals. *Robotics and Autonomous Systems*, 16(2-4) :291–320.

Mataric, M. (1992). Integration of representation into goal-driven behavior-based robots. *IEEE Trans. Robotics and Automation*, 8(3).

Meyer, J. and Wilson, S. (1991). editors : Proceedings of the first international conference on simulation of adaptive behavior - from animals to animats.

Moore, R. (1979). *Methods and Applications of Interval Analysis*. SIAM, Philadelphia.

Pendrith, M. and McGarity, M. (1998). An analysis of direct reinforcement learning in non-markovian domains. *The Fifteenth International Conference on Machine Learning*.

Rasmussen, J. (1986). *Information Processing and Human-Machine Interaction ; An approach to cognitive engineering*. Elsevier Science Publishing Co, North-Holland.

Seward, J. (1949). An experimental analysis of latent learning. *Journal of Experimental Psychologie*, 39 :177–186.

Shannon, C. (1948). A mathematical theory of communication. *Bell System technical Journal*, 27 :379–423,623–656.

Stolzmann, W. (1998). Anticipatory classifier systems. pages 658–664.

Watkins, C. (1989). *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK.

Table des figures

1 Dépendance de l'information perceptive avec la mémoire du système apprenant . xxii
2 Modèle de notre système apprenant xxiii

1.1 Schéma général d'un agent apprenant par renforcement 5
1.2 Algorithme de haut niveau d'un essai d'AR 6
1.3 Cas d'un problème type vérifiant la propriété (P_ϵ) 13
1.4 Principe de la machine de Turing 14
1.5 Hypothèse de modélisation du flux d'événements « non-viable » 17
1.6 Modélisation d'un flux d'erreurs possédant deux sources distinctes 18
1.7 Exemple de fonction de répartition des durées de viabilité, obtenu pour $\epsilon = 10^{-5}$ 20
1.8 Exemple de densité de répartition des durées de viabilité, obtenu pour $\epsilon_1 = 10^{-5}$, $\epsilon_2 = 10^{-2}$ et p=0.5 21
1.9 Coupes du découpage de l'espace d'état pour le problème du pendule inversé . . . 25
1.10 Graphes associés à la sous-section 1.4.4 27
1.11 Graphes associés à la sous-section 1.4.5 29
1.12 Graphes associés à la sous-section 1.4.6 33

2.1 Exemple d'invariant structurel en mécanique. 41
2.2 Les catégories d'états du système. 42
2.3 L'apprentissage est provoqué par l'ajout d'une transition dans le graphe d'états. . 44
2.4 Exemples de cas de multiplicité des solutions au problème de marquage 51
2.5 Exemples de graphes pour le problème du labyrinthe. 60
2.6 Particularités de l'algorithme CbL (1) 62
2.7 Particularités de l'algorithme CbL (2) 63
2.8 Résultats pour l'environnement 1 66
2.9 Résultats pour l'environnement 2 67
2.10 Adaptation à des modifications successives de l'environnement 68
2.11 Le robot miniature Khepera. 70
2.12 Graphes de la sous-section 2.7.3. 71
2.13 Hiérarchisation des agents utilisés par Khepera dans son comportement d'évitement d'obstacles 75
2.14 Comportement de recherche d'objectif avec évitement d'obstacles de Khepera. . . 77
2.15 Résultats d'apprentissage concernant l'agent T_1. 78
2.16 Cas d'échec amenant le robot à s'éloigner du mur. 78
2.17 Nouvel environnement présenté au robot. 79
2.18 Exercices de difficulté progressive proposés au robot. 82

A.1 Architecture pour l'itération de politique 83

A.2 Architecture pour l'itération sur les valeurs 84
A.3 Le problème du pendule inversé 85

3.1 Vue générale du sous-système d'apprentissage perceptif 93
3.2 Définition d'un focus 94
3.3 Sélection d'une hypothèse. 95
3.4 Exemple d'un ensemble S(t) comportant deux hypothèses 98
3.5 Transformée de Hough. 99
3.6 Deux exemples d'application de la contrainte CU. 109
3.7 Mise en évidence d'une solution « absurde » lorsque le nombre d'hypothèses est trop élevé. 112
3.8 Signal composé à partir d'une distribution gaussienne bimodale. 114

4.1 Trois cas de figure, suivant la densité de probabilité associées aux B_j. 121
4.2 Courbes reliant les paramètres h et l permettant la détection fiable d'un signal de bruit gaussien d'amplitude σ 122
4.3 Imprécision sur X_d. 123
4.4 Évolution du taux de détection de l'information perceptive en fonction de l'imprécision sur X_d 124
4.5 Évolution du taux de détection de l'information perceptive en fonction de l'imprécision sur σ 125
4.6 Quatre résultats de l'algorithme de sélection 130
4.7 Évolutions comparées de la probabilité de détection d'information perceptive à partir d'une entrée aléatoire. 131
4.8 Processus de sélection possédant deux signaux d'entrée. 133

B.1 L'intervalle [0,1] vu avec une résolution r=6. 137
B.2 Application du signal X, vue sous différentes résolutions 137
B.3 La suite de signaux donne une tendance de retournement 137
B.4 Segments orientés. 138
B.5 Le signal X est incohérent pour une résolution r=6 138
B.6 Construction d'une suite de segments touchés par un signal 138
B.7 La notion de cohérence est dépendante de la résolution. 139
B.8 Cohérence détectée en fonction de l'amplitude du bruit de mesure 142
B.9 Cohérence détectée en fonction du taux de données aléatoires 143
B.10 Cohérence détectée en fonction du taux d'échantillonnage. 143
B.11 Graphe des évolutions possibles dans le processus de détection d'un segment orienté145
B.12 Confrontation des résultats théoriques et expérimentaux concernant l'expression de Pr- 146
B.13 Confrontation des résultats théoriques et expérimentaux concernant les expressions Prk et Prk+/- 148
B.14 Une idée de l'évolution des probabilités u et u' suivant i. 154
B.15 Évolution de la suite v, suivant m, à h fixé. 155

5.1 Schéma classique du traitement du phénomène d'association perception/action . 174
5.2 Schéma mettant en œuvre un processus d'anticipation 175
5.3 Démarches comparées d'atteinte d'objectif 177

6.1 Principe de la machine de Turing 183

6.2 Une approche fonctionnaliste en sciences cognitives 188
6.3 Une démarche confrontée à la gestion de l'incertitude 189
6.4 Le symbole vu comme auto-entretien d'un signal sur un cycle neuronal 189
6.5 Aspect du fonctionnalisme : nécessité d'un monde pouvant être décrit objectivement 193
6.6 Démarche analytique globale de conception d'un problème complexe 194
6.7 Démarche analytique de conception d'une fonctionnalité 194
6.8 Démarche synthétique de conception d'une fonctionnalité 195
6.9 Démarche synthétique globale 195

Première partie

Apprentissage d'objectif (AO)

1

Influence du contexte sur l'apprentissage par renforcement

1.1 Introduction

1.1.1 Idées directrices

Avant d'entamer ce chapitre, rappelons que notre thèse est fondée sur un **postulat** (voir §2.1, page xviii). Nous avons émis l'hypothèse, dans l'avant-propos, que **les techniques d'apprentissage conventionnelles ne respectent pas les conditions de notre postulat**. En particulier, les conditions de **fiabilité** et de **prédictibilité** ne sont pas garanties en pratique.

Ce chapitre donne quelques arguments en faveur de ce point de vue. Nous centrons notre développement sur la notion de **contexte d'apprentissage**, dont nous supposons qu'il est responsable, en partie, du non-respect de notre postulat.

Nous rassemblons dans le terme « contexte d'apprentissage » l'ensemble des données et modèles utilisés par la méthode d'apprentissage, sans que celle-ci puisse les modifier au cours de son exécution.

Pour étayer notre propos, nous nous focaliserons sur une variante d'une méthode d'apprentissage par renforcement (AR) classique : le Q-Learning. Les méthodes d'AR offrent l'avantage de construire, par essais/erreurs successifs, un mécanisme de réflexe perception/action optimal, sans nécessiter l'utilisation d'un modèle (respect de la première condition de notre postulat) : le système doit seulement connaître à tout moment de l'apprentissage si l'objectif est atteint ou non. Notre choix s'est porté sur cette catégorie de méthodes semi-supervisées, car la phase d'apprentissage de l'atteinte d'un objectif (AO) que nous avons considérée dans la partie introductive de notre document, utilise également un signal de retour pauvre. Nous souhaitons ainsi montrer les problèmes que notre système d'apprentissage à deux sous-systèmes devra éviter.

Le contexte particulier aux méthodes d'AR est composé des éléments suivants :

- la nature et de la qualité des signaux d'entrée du système
- la modélisation du processus de décision par une chaîne de Markov, voire une chaîne de Markov cachée
- la manière d'utiliser les signaux d'entrée pour déterminer l'état courant du système (ce que

nous nommerons *topologie des états du système*)
- la politique d'exploration de l'espace d'états

Chacun de ceux-ci est une cause potentielle d'échec de l'apprentissage. Nous allons illustrer cela à travers un exemple applicatif simple : le problème du pendule inversé. Nous l'avons choisi parce qu'il est un banc d'essai classique en AR et qu'il est particulièrement sensible à la qualité du contexte d'apprentissage : deux ou trois mauvaises commandes consécutives peuvent mettre le système en échec. Nous nous intéresserons plus particulièrement à l'influence du contexte d'apprentissage sur la fiabilité du système.

1.1.2 Principaux résultats

Une première conclusion de notre étude est que les caractères de fiabilité et de prédictibilité ne peuvent pas être appliqués sur une modélisation d'un problème simple (celui du pendule inversé) par une technique d'apprentissage par renforcement.
Cependant, nous ne limitons pas notre travail à une simple constatation. Nous essayons de préciser les causes possibles de cette constatation. Pour cela, nous allons considérer le graphe d'états du système (chaque transition correspond au passage du système d'un état e_i à un état e_j par une commande a_k). Nous mettons en évidence la relation entre la dégradation des résultats du système et l'augmentation du nombre de transitions différentes, partant du même état e_i en exécutant la même commande a_k. Cette dégradation a également un rapport avec l'augmentation du nombre de transitions d'un état e_i vers un état e_j (l'exécution de plusieurs commandes aboutit au même état). Ces deux phénomènes ont un rapport direct avec la quantité d'information (au sens de Shannon) portée par l'exécution d'une commande, connaissant l'état courant (pouvoir discriminant des commandes sur la nature de l'état d'arrivée[1]) et avec la quantité d'information portée par la connaissance de l'état d'arrivée (pouvoir discriminant de la connaissance de l'état d'arrivée pour déterminer l'action a_k qui vient d'être exécutée). Nous utiliserons donc naturellement deux mesures d'entropie H_1 et H_2 pour quantifier la qualité du contexte d'apprentissage et donner une relation avec le degré de fiabilité du résultat d'apprentissage.

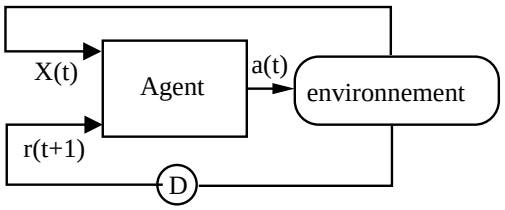
Nous montrons également que certains facteurs du contexte d'apprentissage peuvent être modélisés comme des sources d'erreur, caractérisées par une fréquence d'occurrence des erreurs propre à ces sources.

L'ensemble de cette étude nous permet donc de mieux comprendre par quels mécanismes le contexte influence le résultat de l'apprentissage. Ces résultats nous sont utiles pour définir ce que pourrait être un **contexte idéal** (minimisant H_1 et H_2).

1.1.3 Guide du chapitre

La section 1.2 donne une brève présentation du mécanisme de l'AR et des méthodes de résolution, puis fait apparaître les causes d'incertitude inhérentes aux algorithmes d'AR. Nous précisons le rapport entre fiabilité et prédictibilité d'une part[2], et incertitude et imprécision d'autre part. Dans la section suivante, nous décrivons les moyens d'étude de l'incertitude liée au contexte d'apprentissage. Dans un premier temps, nous proposons d'utiliser des mesures d'entropie de Shannon pour apprécier l'incertitude apportée par le contexte d'apprentissage ; elles quantifient le caractère discriminant de l'exécution d'une commande sur la nature de l'état

1. L'état d'arrivée est l'état e_j obtenu après l'exécution de la commande a_k à partir de l'état e_i 2. La signification que nous donnons à ces deux termes a été spécifiée dans l'avant-propos.



À chaque pas de temps t, l'agent prend connaissance du vecteur d'entrée X(t) et délivre une action a(t) en fonction de X(t). Au pas de temps suivant, la réaction de l'environnement provoque l'envoi du signal de renforcement r(t+1), dont la valeur dépend de la qualité de l'action a(t) par rapport à un objectif fixé *a priori*.

Fig. 1.1 – Schéma général d'un agent apprenant par renforcement

résultant. Dans un deuxième temps, nous proposons une modélisation statistique du résultat de cette incertitude en le caractérisant grâce à un flux d'événements[1] dont on peut déterminer les caractéristiques (fréquence moyenne d'apparition). Enfin, la dernière section décrit une série d'expériences autour du problème du pendule inversé : nous souhaitons y montrer l'influence du contexte de l'apprentissage sur les performances du système apprenant, indépendamment de la méthode d'apprentissage elle-même. Nous examinons la relation entre l'évolution des mesures et la dégradation de la fiabilité du système ; d'autre part, nous regardons dans quelle mesure le manque de fiabilité peut être modélisé statistiquement par un flux d'erreurs de paramètres constants.

1.2 Cadre général de l'apprentissage par renforcement (AR)

1.2.1 Le mécanisme d'AR schématisé - les termes clés

Le contexte général de l'AR est l'apprentissage de la « meilleure » association possible perception/action, suivant un objectif donné. Cet apprentissage est guidé par un signal pauvre, appelé *signal de renforcement*, qui indique à tout moment la qualité de l'action qui vient d'être exécutée par rapport à l'objectif. Ce signal peut être pauvre et indiquer simplement si l'objectif est atteint (renforcement positif), n'est pas atteint (renforcement nul) ou si un événement dommageable s'est produit (renforcement négatif). La figure 1.1 schématise le système d'AR.

L'agent subit un apprentissage tout au long de sa vie. L'objectif est de pouvoir faire face à un changement éventuel de l'environnement et de s'y adapter (caractéristiques d'incrémentalité) : les algorithmes d'AR sont donc itératifs et sont utilisés « en ligne ». Des termes clé seront utilisés par la suite. Nous les définissons ici :

- une *itération* de l'algorithme d'AR correspond au réflexe perception/action décrit dans le schéma 1.1, pour un pas de temps. Elle comprend également l'adaptation des paramètres de l'agent apprenant face à la donnée du signal de renforcement.
- un *essai* est une suite d'itérations de l'algorithme d'AR, commencée en fixant l'état initial de l'agent et terminée lorsque le signal de renforcement prend une certaine valeur (par exemple, l'objectif est atteint) ou qu'une durée arbitrairement choisie avant l'expérience est atteinte. L'algorithme de haut niveau est donné par la figure 1.2.

1. Les événements qui nous intéresserons dans la section applicative seront les sorties du système de sa zone de viabilité.

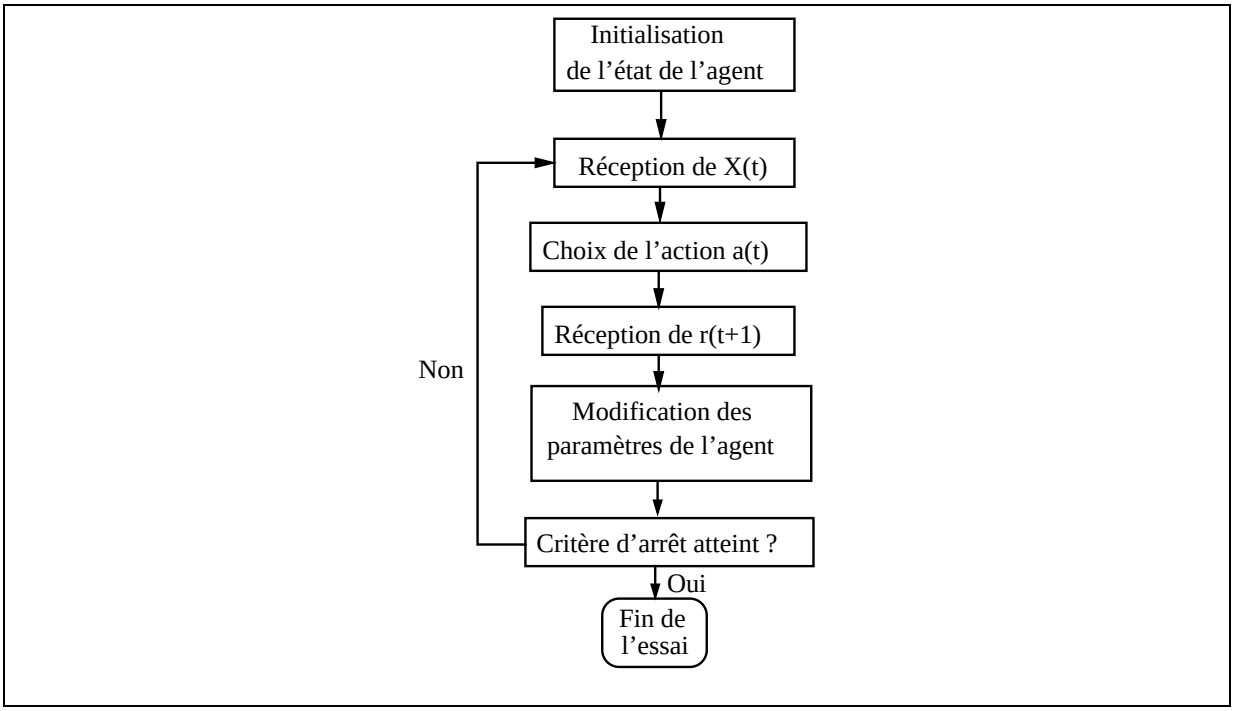


FIG. 1.2 – Algorithme de haut niveau d'un essai d'AR

– la durée d'apprentissage est l'ensemble des essais nécessaires pour atteindre l'objectif. Si, au bout d'un certain nombre d'essais (fixé arbitrairement *a priori*), l'objectif n'est pas atteint, on considère que l'apprentissage a échoué.

Les paramètres de l'agent sont initialisés au début de la phase d'apprentissage et sont modifiés à chaque itération, sans être réinitialisés au début de chaque essai.

1.2.2 Références générales

Nous ne souhaitons pas aborder l'étendue de la problématique posée par l'AR, qui est un domaine de recherche vaste, en constante évolution. Pour une information générale sur ce domaine, le lecteur pourra consulter avec profit les références suivantes :

– pour une première approche, on pourra lire l'état de l'art établi par Kaelbling [Kaelbling et al., 1996]
– le livre écrit par Sutton est une introduction très complète [Sutton et Barto, 1998]. La thèse de Wiering est également très riche [Wiering, 1999]
– pour une formalisation plus poussée et le détail des preuves de convergence de l'AR dans le cas discret, on pourra consulter [Bersekas et Tsitsiklis, 1996], alors que la thèse de [Munos, 1997] fournit des preuves de convergence dans le cas continu.

1.2.3 Positionnement de l'AR par rapport à d'autres méthodes d'apprentissage

La caractéristique majeure de l'ensemble des techniques que l'AR englobe est que l'apprentissage est semi-supervisé. Le terme « semi-supervisé », par opposition à « supervisé » ou « non-supervisé », désigne la nature de l'information qui est transmise au système apprenant à chaque pas de temps en retour de l'action qu'il vient d'accomplir. Voici une brève explication des deux termes « supervisé » et « non-supervisé », ainsi que les outils les plus courants qui leur sont

associés.

Dans le cas « supervisé », cette information correspond à l'erreur exacte entre ce que le système a accompli et ce qu'il aurait dû faire ; cela suppose la connaissance *a priori* d'un ensemble d'exemples d'associations perception/action. L'apprentissage revient alors à construire un modèle général à partir de ces exemples, supposés être représentatifs du fonctionnement du système. Donc, d'un point de vue pratique, il s'agit de déterminer les paramètres d'une fonction interpolant « au mieux » cette base d'exemples perception/action. Si on possède uniquement des informations numériques sur le processus, on pourra utiliser des réseaux de neurones multicouches (dont les fonctions de base sont des sigmoïdes[1]) ou des réseaux RBF[2] (dont les fonctions de base sont typiquement des gaussiennes). Par contre, si on possède une information plus qualitative, on pourra utiliser des techniques de logique floue (systèmes d'inférence floue).

Dans le cas « non-supervisé », une loi d'évolution, fixée *a priori*, dirige la modification des paramètres du système apprenant. Les cartes auto-organisatrices de Kohonen en sont un exemple. L'idée est ici de créer en ligne un découpage de l'espace, induit par les variables d'entrée du système en un ensemble de zones élémentaires (assimilables à un pavage de Voronoï) qui sont chacune sous l'influence d'un vecteur paramètre (neurone). Les vecteurs paramètres sont modifiés à chaque pas de temps suivant une loi barycentrique entre le vecteur d'entrée observé et le vecteur paramètre le plus proche, ainsi que ses voisins (spécifiés par une certaine topologie définie *a priori*). Ces zones élémentaires peuvent être regroupées en classes *a posteriori* grâce à un algorithme supervisé. Le principe d'évolution des paramètres permet de resserrer le maillage autour des zones de l'espace d'entrée ou un maximum de points se concentrent.

L'apprentissage semi-supervisé est, comme son nom l'indique, un intermédiaire entre les deux approches précédentes. L'idée principale est de guider l'apprentissage, à chaque pas de temps, en mesurant la qualité de l'action entreprise au pas de temps précédent, par rapport à l'atteinte d'un objectif donné ; cette mesure est donnée par un **signal de renforcement r(t)**, que l'on peut comparer à une récompense (si l'objectif est atteint) ou à une punition (une erreur est commise). La valeur de ce signal peut être très pauvre, voire binaire (atteinte oui ou non de l'objectif à l'instant t). L'objectif de l'apprentissage est d'estimer, par essais/erreurs successifs, la qualité de chaque action dont le système dispose à l'instant t, en fonction des données d'entrées (perception) et de l'objectif à atteindre, afin de choisir la meilleure.

1.2.4 Problématique et méthodes de résolution liées à l'AR

Le problème que nous avons brièvement explicité ci-dessus est un problème d'optimisation. Pour un vecteur d'entrée donné X(t) (perception) à l'instant t, il s'agit de choisir à tout instant t l'action a(t) qui maximise l'espérance de la somme des renforcements r(t+k) futurs (dans un horizon de temps fini ou infini). La technique mathématique sous-jacente est la programmation dynamique (voir l'article de référence [Bellman, 1957] ou, pour une vue générale, le livre de Bertsekas [Bertsekas, 1987]) : cela n'est pas surprenant car le problème de maximisation proposé ci-dessus est décomposable en un problème à l'instant t (choisir la meilleure action à l'instant t) et en un autre problème (maximiser l'espérance des renforcements futurs à l'instant t+1) ; donc, la connaissance, par récursions successives, des différents sous-problèmes à l'instant t+k, puis t+k-1, ..., puis t+1 permet finalement de résoudre le problème à l'instant t. Mais, celle-ci

1. Une fonction sigmoïde est une fonction à valeurs dans $\mathbb{R}$, strictement monotone et bornée. 2. RBF signifie Radial Basis Function.

n'est appliquable que si on peut connaître parfaitement, à l'instant t, l'évolution du système par application d'une séquence quelconque d'actions.

Historiquement, l'AR s'est répandue grâce aux travaux précurseurs de Barto et Sutton au début des années 1980 (voir l'article de référence [Barto et al., 1983] sur la méthode AHC[1]), puis de Watkins ([Watkins, 1989]) donnant naissance à la technique du Q-Learning. Ces deux techniques sont les archétypes de deux classes d'algorithme d'AR : les algorithmes fonctionnant par itération sur la politique d'actions (AHC) et les algorithmes cherchant la fonction valeur optimale (Q-Learning, Sarsa [Rummery, 1995], R-Learning [Schwartz, 1993]), celle-ci influençant directement le choix de l'action a(t) à tout moment. L'architecture ainsi que l'algorithme de haut niveau associé au Q-Learning est disponible à l'annexe A.1.2. Le schéma de haut niveau de l'algorithme AHC se trouve en annexe (figure A.1, page 83).

Les méthodes d'AR basées sur la recherche de la fonction valeur optimale sont les plus répandues, car elles sont en fait plus simples d'emploi. Dans ce cadre, le problème est d'évaluer cette fonction valeur. L'idée est de transformer les équations du problème de programmation dynamique de manière à résoudre itérativement le problème de PD : cette écriture itérative se nomme la technique des différences temporelles (TD)[2] (voir [Sutton, 1988] pour TD). Cette méthode permet d'estimer la fonction valeur, sans pour autant connaître un modèle de l'environnement. D'autres méthodes découlent de TD : on peut citer TD(λ), qui introduit une « trace d'éligibilité » qui va permettre d'influencer la valeur non plus du dernier état parcouru (à l'instant t), mais l'ensemble des valeurs des états passés, avec toutefois une atténuation exponentielle de cette modification (donnée par la valeur[3] du paramètre λ). L'algorithme $Q(\lambda)$ est une extension du Q-Learning original, utilisant cette trace d'éligibilité ([Peng et Williams, 1996]).

Pour mettre en oeuvre les techniques d'AR, il faut modéliser ce qui va constituer la « mémoire » du système, c'est-à-dire le support de la fonction valeur. Voici plusieurs possibilités qui ont été développées dans la littérature, dans le cas où l'espace d'états est continu[4] et où il est confondu avec l'espace engendré par les signaux d'entrée du système :

- découpage de l'espace des variables d'entrée en un ensemble fini de boîtes [Barto et al., 1983]
- interpolation linéaire de la fonction valeur
- interpolation de la fonction valeur par un réseau de neurones multi-couches ([Sutton, 1984], [Sutton, 1988])
- utilisation de réseaux CMAC[5] ([Lin et Kim, 1991],[Sutton, 1996])
- utilisation de réseaux RBF
- utilisation de systèmes d'inférence floue ([Nowé, 1995])
- maillage par triangulation de Delauney ([Munos, 1997])

1.2.5 Conditions de convergence des algorithmes d'AR

D'un point de vue théorique, des preuves de convergence de l'algorithme vers une politique de choix d'action optimale n'ont été obtenues que pour les algorithmes utilisant la méthode des

1. AHC signifie Adaptive Heuristic Critic 2. Historiquement, elle a été employée pour la première fois par Samuel dans le cadre d'un programme d'échecs auto-apprenant [Samuel, 1959] 3. Lorsque cette valeur est nulle, on retrouve le cas particulier de TD, noté également TD(0). 4. Si l'espace d'états est discret, la fonction valeur se résume en un ensemble fini de valeurs réelles et le problème ne se pose plus. 5. CMAC signifie Cerebellum Model Articulation Controller . Les articles de référence sont [Albus, 1971] et [Albus, 1981]

différences temporelles (Q-Learning, Sarsa), en faisant l'hypothèse que le processus de décision est modélisable par une chaîne de Markov (MDP), possédant un nombre d'états fini. Le cas où l'espace d'états du système est discret a été traité en premier : le lecteur pourra consulter [Sutton, 1988] et [Dayan, 1992] pour les preuves de convergence de TD(0), puis [Dayan, 1992], [Dayan et Sejnowski, 1994] et [Tsitsiklis, 1994] pour les preuves de convergence de TD(λ). Dans le cas où l'espace d'états est continu, la preuve a été apporté par [Munos, 1997], par découpages successifs de l'espace d'états (triangulation de Delauney).

Dans les autres cas, cités dans la sous-section précédente, il n'y a aucune garantie théorique de convergence, ce qui ne signifie pas que l'apprentissage ne fonctionne pas : historiquement, l'exemple le plus spectaculaire est fourni par le « joueur » de backgammon de Tesauro, utilisant une modélisation de la fonction valeur par un réseau de neurones multi-couches [Tesauro, 1994]. Mais, les (nombreux) paramètres peuvent être délicats à fixer, même pour des problème apparemment « simples », en particulier lorsque la fonction valeur possède des discontinuités importantes [Bersini et Gorrini, 1996]. [Tsitsiklis et Roy, 1996] montre que, dans le cas d'une fonction d'approximation non-linéaire, la méthode TD peut devenir instable. Il en est de même pour le Q-Learning avec fonction d'approximation linéaire [Baird, 1995]. Pendrith souligne aussi la possible difficulté d'appliquer les algorithmes d'AR en robotique mobile ([Pendrith et McGarity, 1998], [Pendrith, 1999]).

1.2.6 Causes de difficultés dans l'utilisation des algorithmes d'AR

Dans les sous-sections précédentes, nous avons identifié les mécanismes de haut niveau des algorithmes d'AR et présenté le cadre général pour lequel des preuves de convergence ont été établies. En pratique, des difficultés peuvent apparaître dans les cas suivants :

- le temps pour obtenir une solution convenable est prohibitif
- le problème de décision n'est pas markovien
- la fonction valeur est difficile à interpoler

Nous allons développer brièvement les deux premiers points. Pour le dernier, le lecteur pourra se reporter à [Bersini et Gorrini, 1996].

Le premier point est lié au mécanisme d'exploration de l'espace d'états du système, intervenant au moment de choisir l'action à exécuter à l'instant t. Cette problématique est vaste et c'est un des enjeux actuels majeurs dans le domaine de l'AR : le lecteur pourra se reporter à [Wiering, 1999] pour un point de vue complet. En bref, si la nature du problème implique que la probabilité d'atteindre au hasard une solution relativement correcte (qu'on pourra optimiser par la suite) est très faible, le temps (qu'on peut compter en nombre d'essais) pour atteindre cette première solution peut être très important, rendant l'algorithme inutilisable en pratique. Or, dans un problème complexe, l'espace d'états est très souvent grand. Il faut donc guider l'apprentissage. Trois catégories de solutions peuvent être apportées pour réduire cette difficulté :

- découper le problème en sous-problèmes, hiérarchiquement connectés [McGovern et al., 1998], [Davesne et Barret, 1999]
- introduire de la connaissance *a priori* pour éliminer immédiatement des possibilités visiblement non intéressantes (les travaux portant sur le Q-Learning flou vont dans ce sens [Jouffe, 1997])
- modifier la topologie du modèle du système de manière à ne conserver que les états « importants » du système (manière dont on construit les états du système) : les méthodes

procédant par regroupement d'états vont dans ce sens.

L'hypothèse markovienne est théoriquement essentielle, puisque les résultats de convergence connus jusqu'à présent ne s'appliquent que sur des MDP. Nous rappelons ici que l'hypothèse markovienne impose que la probabilité $p_{i,j}$ de passage d'un état i à un état j ne dépend que de i ; cela signifie que les états du système antérieurement à son arrivée dans i n'interviennent pas dans le calcul de $p_{i,j}$. Pratiquement, il est difficile de savoir *a priori* si le problème, en réalité, est ou n'est pas modélisable par un MDP. En effet, cela dépend principalement de deux catégories de facteurs :

- la pertinence et la « précision » des données d'entrée
- la pertinence du choix de la topologie des états par rapport au choix et à la qualité des données d'entrée, mais aussi par rapport à l'objectif de l'apprentissage

Le premier point est évident : les données d'entrée doivent être utilisable pour déduire les états du système, directement (modélisation par une chaîne de Markov) ou indirectement (modélisation par une chaîne de Markov cachée [1]). Le second est moins immédiat. Il traduit la capacité de l'expérimentateur à créer un lien entre la réalité de l'expérience et le modèle théorique dont l'algorithme a besoin pour fonctionner correctement : ce lien se traduit précisément par la construction d'un contexte permettant ce bon fonctionnement ; il utilise des connaissances *a priori*. Prenons l'exemple du problème du pendule inversé pour préciser notre pensée. Si l'objectif est de maintenir le pendule proche de l'équilibre et le chariot proche de son point d'origine, on fera en sorte que la majorité des états du système se trouvent dans cette zone d'équilibre. Mais le degré de cette concentration de l'information est contrebalancé par la qualité des données d'entrée du système (position angulaire et vitesse angulaire de la tige, position et vitesse du chariot) : un bruit de mesure trop important rend inefficace une politique de regroupement de l'information, car il existe alors trop d'incertitude sur l'état dans lequel se trouve réellement le système, donc sur l'action correcte à exécuter.

Nos remarques soulignent l'importance du contexte de l'algorithme d'AR, c'est-à-dire de ce qui appartient au savoir-faire de l'expérimentateur. Ce contexte permet de rapprocher la réalité expérimentale et le modèle théorique et tourne autour de la spécification des états du système.

1.2.7 Lien entre incertitude, imprécision, fiabilité et prédictibilité

Le terme « incertitude » est très vague en lui-même. Dans notre cas, les points suivants sont *a priori* sujets à l'incertitude :

- l'état du système, à un instant donné
- la transition d'un état supposé connu à un autre, à un instant donné
- la validité du choix d'une action ou d'une hypothèse, à un instant donné, suivant un objectif précis connu *a priori* (reconnaître correctement un phonème, effectuer une séquence d'actions atteignant l'objectif souhaité)
- la validité du modèle choisi pour le système (le modèle Markovien est-il adapté au problème réel ?)

L'imprécision concerne, elle, l'imperfection des données d'entrée du système. Si on suppose qu'on n'a aucun contrôle sur la qualité des données d'entrée du système, l'imprécision est une source d'incertitude. L'algorithme d'apprentissage vise à lever l'incertitude sur la validité du choix d'une action lorsqu'on connaît l'état du système. Dans le cas de l'AR, la modélisation du

1. La classe de problème est alors nommée POMDP pour *Partially Observable Decision Process*. On pourra consulter [Lovejoy, 1991] ou [Wiering, 1999]

processus de décision par une chaîne de Markov ou une chaîne de Markov cachée signifie qu'on ne cherche pas à lever l'incertitude sur l'état obtenu après l'exécution d'une action à partir d'un état supposé connu. Enfin, l'AR n'a aucune prise sur l'incertitude sur l'état courant, ni sur celle provoquée par la topologie des états.

Nous voyons donc que l'apprentissage en lui-même ne lève qu'une partie des incertitudes. Il s'ensuit que la fiabilité du processus obtenu, ainsi que la prédictibilité du résultat de l'apprentissage en sont affectés.

1.3 Outils d'étude de l'incertitude due au contexte de l'AR

1.3.1 Introduction

Dans la suite de ce document, nous modéliserons le contexte d'AR par un graphe d'états dont chaque transition est associée à l'exécution d'une commande particulière. Dans ce cadre, nous présentons un outil de mesure de l'incertitude classique : la mesure d'entropie de Shannon. Il s'agit de mesurer la possibilité de prévoir le résultat de l'exécution d'une commande (l'état suivant), connaissant l'état présent (grâce à la mesure H_1), mais aussi de mesurer la possibilité de déduire la commande exécutée connaissant les deux derniers états du système (grâce à la mesure H_2). Cette mesure induit une définition de la qualité du contexte d'apprentissage, ainsi qu'un contexte idéal (minimisant les deux mesures). La mesure de la qualité du contexte d'apprentissage peut se faire après une exploration des états du système, établissant ainsi les différentes transitions du graphe d'états. Elle peut être effectuée avant l'apprentissage lui-même (en choisissant aléatoirement, pour chaque état, la commande à exécuter). Nous montrerons, dans la section suivante, dans quelle mesure elle peut servir d'indicateur *a priori* sur la fiabilité de la politique de commande obtenue après l'apprentissage.

Nous souhaitons également caractériser la fiabilité de la politique de commande après l'apprentissage. Nous pensons que certains éléments du contexte d'apprentissage **perturbent aléatoirement** cette fiabilité, en entraînant une production aléatoire d'événements fâcheux (non atteinte de l'objectif, ou sortie de la zone de viabilité). Dans le cadre d'un problème de viabilité, la fiabilité est en relation avec la durée de viabilité du système. Celle-ci peut donc être vue comme la réalisation d'une variable aléatoire, dont nous allons proposer une loi de probabilité. Cette loi utilise un paramètre, dont nous proposerons un estimateur.

1.3.2 Qu'entendons-nous par « qualité » du contexte d'apprentissage ?

Un algorithme d'AR utilise des données provenant d'un contexte sur lequel il ne peut pas agir. Ce contexte est formé principalement des éléments suivants :

- la topologie des états du système
- la dynamique du système
- la nature et la qualité des données d'entrée
- les actions du système lui permettant de changer d'état
- les paramètres internes à l'algorithme, que celui-ci ne peut pas modifier au cours de son exécution

Nous pensons que l'adéquation entre la topologie des états du système, la dynamique du système, la nature et la qualité des données d'entrée et les actions du système est nécessaire

à l'obtention d'un bon résultat d'apprentissage. Nous proposons un moyen de quantifier cette adéquation, utilisant l'entropie associée aux résultats possibles de l'exécution d'une action, c'est-à-dire aux différents états auxquels le système peut parvenir après avoir exécuté une action, connaissant l'état au moment de l'exécution de celle-ci. En cela, nous regardons l'action comme une faculté discriminante de la perception (état du système). Cette idée est une hypothèse développée dans le cadre du vivant : nous en reparlerons dans la troisième partie de ce document de thèse.

La « qualité » du contexte d'apprentissage est représentée par la mesure de l'adéquation présentée ci-dessus. L'objet de cette section est de spécifier cette mesure, qui est utilisée classiquement en théorie de l'information et en théorie de la décision.

1.3.3 Notations

On considère un ensemble de n états e_1,e_2,...,e_i,...,e_n, ainsi qu'un ensemble de q actions a_1,a_2,...,a_k,...a_q. À chaque instant t, le système se trouve dans un état e_i et choisit d'exécuter une action a_k jusqu'à ce qu'il se trouve dans un état e_j différent de e_i, à l'instant t+h. On considère également l'état transitoire $e_{i,k}$ dans lequel le système se trouve entre les instants t et t+h. Il faut donc distinguer l'état e_i du système alors qu'il n'a pas fait son choix sur l'action à exécuter, l'état $e_{i,k}$ pour lequel le système a fait son choix à partir de l'état e_i et l'état e_j qui résulte de l'exécution de a_k à partir de e_i.

Les probabilités pour que le système se trouve respectivement dans l'état e_i et $e_{i,k}$ seront notées p_i (resp. $p_{i,k}$). Pour un état e_i et une action a_k fixés, on considère l'ensemble des probabilités $p_{i,k,j}$ d'atteindre les autres états du système, à l'exception de e_i, grâce à l'action a_k. Nous avons la relation :

$$\forall i \in \{1,...,n\}, k \in \{1,...,q\}, \quad \sum_{j \in \{1,..,n\}, j \neq i} p_{i,k,j} = 1$$

1.3.4 Contexte idéal et quasi-idéal - Propriété (P_ϵ)

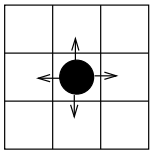
Nous supposons que le contexte idéal possède la propriété (P) suivante :

> *Pour tout état e_i du système et toute action a_k disponible lorsque le système est dans l'état e_i, toutes les probabilités $p_{i,k,j}$ sont nulles à l'exception d'une seule qui vaut 1. De même, pour tout état e_i du système, si une action a_k amène celui-ci dans l'état e_j, alors aucune autre action ne peut l'amener dans l'état e_j.*

Cela signifie qu'à partir de l'état e_i, il est possible de prédire avec une parfaite exactitude le résultat de l'action a_k sur l'évolution de l'état du système et que si on connaît deux états consécutifs e_i et e_j, alors on sait en déduire avec exactitude l'action exécutée à partir de l'état e_i

En marge de ce contexte idéal, nous allons plus particulièrement nous intéresser à un ensemble de contextes quasi-idéaux, respectant la propriété (P_ϵ) suivante :

> *On considère une valeur réelle ϵ strictement positive et très proche de 0. Pour tout état e_i du système et toute action a_k disponible lorsque le système est dans l'état e_i, toutes les probabilités $p_{i,k,j}$ sont inférieures à ϵ à l'exception d'une seule qui est*



Chaque action mène à un état unique (c'est-à-dire à une case unique) et, si on connaît la case d'arrivée, on peut déterminer sans ambiguïté l'action qui vient d'être exécutée.

FIG. 1.3 – Cas d'un problème type vérifiant la propriété (P_ϵ)

supérieure à $1 - q\epsilon$. *De même, pour tout état* e_i *et tout état consécutif* e_j *du système, les probabilités* $p_{i,k,j}$ *sont toutes inférieures à* ϵ *sauf une qui est supérieure à* $1 - n\epsilon$.

Si ϵ est suffisamment petit, le contexte quasi-idéal se comporte en pratique (dans la durée de l'expérience) comme le contexte idéal : la probabilité de découvrir deux transitions différentes à partir d'un même état transitoire $e_{i,k}$ est si faible, que la réalisation de cet événement n'arrive pas dans la durée de l'expérience. De même, la probabilité pour que deux actions différentes exécutées à partir d'un état e_i amènent au même état e_j est si faible que cela n'arrive pas en pratique.

Dans la suite de notre discours, nous confondrons les propriétés (P) et (P_ϵ) en utilisant uniquement le terme « (P_ϵ) ».

1.3.5 Exemples de contextes vérifiant ou ne vérifiant pas (P_ϵ)

Le problème type respectant (P_ϵ) est celui du labyrinthe, découpé en un ensemble de cases, pour lequel les actions permises sont de se déplacer sur une case adjacente à la case courante (voir la figure 1.3). Dans un cadre plus général, une machine de Turing respecte (P_ϵ) car l'évolution de l'état du système est régie par un ensemble de règles, permettant pour chaque état du système de déterminer l'état suivant à partir de l'action exécutée (voir la figure 1.4).

Par contre, la manière dont nous construisons les états dans le problème du pendule inversé, proposé dans la section suivante, engendre un contexte d'apprentissage qui ne respecte pas (P_ϵ), même si les données d'entrée sont parfaites.

1.3.6 Information associée à l'exécution d'une action

Nous souhaitons mesurer l'information transmise par le passage d'un état à un autre grâce à une certaine action. Beaucoup d'outils de mesure existent. Leur utilisation dépend de la nature de l'information à traiter. D'après [Bouchon-Meunier, 1989], celle-ci est de deux ordres :

- l'information d'observation, dont la mesure revient à évaluer la qualité des données d'entrée du système
- l'information d'exploitation, qui permet de prendre une décision. Le terme « information » doit être pris ici au sens de Shannon [Shannon, 1948]

Dans notre cas, les deux types d'informations sont mixés : le résultat de l'exécution d'une action est entaché d'incertitude, du fait de la méconnaissance *a priori* de l'état dans lequel le système va se trouver après l'exécution de cette action, mais aussi parce qu'il existe une imprécision due aux mesures qui rend l'exactitude de la connaissance de l'état courant incertaine. Nous ne souhaitons pas mesurer l'imprécision mais les conséquences de cette imprécision sur l'état du

L'automate de Turing possède un nombre d'états fini et est composé d'une tête devant laquelle se déroule un ruban divisé en cellules et qui peut se déplacer aussi bien de droite à gauche que de gauche à droite. N'importe quelle fonction calculable peut être calculée par cet automate grâce à l'ensemble de règles auxquelles il est soumis et que l'on peut toujours déterminer. Ces règles sont de trois types:

1) Si Aj Bm Sk
Dans un état Si, l'automate remplace Aj, contenu dans la cellule au dessous de la tête par Bm, puis passe dans un état Sk

2) Si Aj DSk
Le ruban est déplacé d'une cellule vers la droite et l'automate passe à l'état Sk

3) Si Aj GSk
Le ruban est déplacé d'une cellule vers la gauche et l'automate passe à l'état Sk

Fig. 1.4 – Principe de la machine de Turing

système à l'instant t+h.

Le choix du type de mesure dépend de la nature de l'information à traiter. La théorie des modèles entropiques permet de classer différents types de mesure pour en extraire une classification des mesures existantes. Elle considère deux types de modèles [Bouchon, 1988] :

- les modèles entropiques de type 1, qui peuvent être considérés pour mesurer l'information contenue dans les résultats d'une expérience soumise à l'incertitude due aux observations. Ces modèles expriment un point de vue informationnel
- les modèles entropiques de type 2, que mesurent l'incertitude intervenant durant l'observation des résultats d'une expérience. Ces modèles expriment une étude de l'imprécision

Notre problématique nécessite l'utilisation d'un modèle de type 1. Parmi ceux-ci, on peut citer l'entropie de Shannon [Shannon, 1948], l'information de Hartley [Hartley, 1928] et la divergence de Kullback-Leibler. L'entropie de Shannon est utilisée comme critère dans l'élaboration d'arbres de décision (algorithme ID3, par exemple [Quinlan, 1984]).

1.3.7 Mesures utilisant l'entropie de Shannon

Nous allons nous intéresser à l'information qui peut être dégagée de l'exécution d'une action a_k. Intuitivement, on peut relier le fait que a_k agit sur le système avec le changement d'état du système. Dans le cas général, ce changement d'état est soumis à l'incertitude, dans le sens où si on connaît parfaitement e_i et a_k, il se peut qu'on puisse ne pas prédire avec certitude la nature de l'état e_j obtenu à l'instant t+h. C'est le sens des probabilités $p_{i,k,j}$ lorsque i et k sont fixés. De même, si on connaît parfaitement deux états par lequel le système est passé consécutivement, on peut ne pas déduire quelle action a_k a permis le passage d'un état à l'autre : c'est le sens des

probabilités $p_{i,k,j}$ lorsque i et j sont fixés.

Par conséquent, nous avons deux mesures différentes, dont l'objectif est de donner une idée de l'incertitude associée soit à la prédiction de l'état futur, soit à la déduction de l'action exécutée par le système. L'utilisation de l'entropie semble donc adaptée à notre problématique. Pour un état e_i et une action a_k fixés, nous définissons l'entropie $H(e_i,a_k)$ associée à l'incertitude sur l'état suivant comme suit :

$$H(e_i,a_k) = - \sum_{j \in \{1,...,n\}, j \neq i} p_{i,k,j} log(p_{i,k,j})$$

De même, pour un état e_i et un état e_j, nous spécifions l'entropie $H(e_i,e_j)$ associée à l'incertitude sur l'action qui a permis le passage de l'état e_i à l'état e_j.

$$H(e_i,e_j) = - \sum_{k \in \{1,...,q\}, j \neq i} p_{i,k,j} log(p_{i,k,j})$$

L'étude détaillée de la fonction H a été effectuée pour la première fois par Shannon dans le cadre de la théorie de l'information [Shannon, 1948]. À partir de cette étude, nous savons que $H(e_i,a_k)$ est minimale et vaut 0 dans le cas où un unique $p_{i,k,j}$ est non nul et vaut 1 (prédictibilité parfaite du résultat de l'action a_k). À l'autre extrême, $H(e_i,a_k)$ est maximale lorsque les $p_{i,k,j}$ sont tous égaux (incertitude maximale sur le résultat de l'action a_k). De même, $H(e_i,e_j)$ est minimale et vaut 0 lorsqu'il existe une action permettant de passer de l'état e_i à l'état e_j. Au contraire, $H(e_i,e_j)$ est maximale lorsque touts les actions ont une probabilité égale d'avoir permis le passage de l'état e_i à l'état e_j.

On remarque que $H(e_i,a_k)$ est minimale et nulle lorsque l'action a_k est parfaitement discriminante à partir de l'état e_i : l'ensemble des probabilités $p_{i,k,j}$ sont nulles sauf une qui vaut 1 (lorsque i et k sont fixés). De même, $H(e_i,e_j)$ est minimale lorsque les états sont parfaitement discriminants par rapport aux actions. Lorsque ces deux cas sont réunis, l'état e_i vérifie la propriété (P).

À partir des définitions de $H(e_i,a_k)$ et de $H(e_i,e_j)$ pour un état particulier e_i, peut-on mesurer le caractère informatif du système total (incluant tous les états et toutes les actions possibles) ? Avant de répondre à cette question, trois contraintes existent :

- la mesure sur le système total doit être minimale lorsque le contexte de l'apprentissage vérifie la propriété (P)
- la mesure sur le système total ne doit tenir compte que des états qui sont effectivement explorés
- la mesure du système total ne doit pas être influencée par l'apprentissage

À partir de la première contrainte, l'expression générale de la mesure de la qualité de discrimination des actions est la suivante :

$$H = \sum_{i \in \{1,...,n\}, k \in \{1,...,q\}} \alpha_{i,k} H(e_i,a_k)$$

avec :

$$\forall i \in \{1,...,n\}, k \in \{1,...,q\},\ \alpha_{i,k} \in [0,1]$$

et

$$\sum_{i\in\{1,\ldots,n\},k\in\{1,\ldots,q\}} \alpha_{i,k} = 1$$

La dernière contrainte signifie qu'on souhaite pouvoir mesurer la qualité du contexte de l'apprentissage seul. Il ne faut donc pas utiliser l'estimation des $p_{i,k}$ pour pondérer les $H(e_i,a_k)$, car ces valeurs dépendent de la politique de commande du système (certains états ne seront quasiment jamais visités, alors que d'autres le seront fréquemment).

Une mesure H_1 peut être envisagée en donnant autant de poids à chaque $H(e_i,a_k)$: on aura $\alpha_{i,k} = 1/(n.q)$. Il vient :

$$H_1 = \frac{1}{n.q} \sum_{i\in\{1,\ldots,n\},k\in\{1,\ldots,q\}} H(e_i,a_k) \tag{1.1}$$

Pour répondre à l'exigence de la deuxième contrainte, on pourra modifier le facteur n.q de manière à ne compter que les états $e_{i,k}$ effectivement atteints.

Une mesure H_2 est obtenue pareillement, avec $\alpha_{i,k} = 1/(n.(n-1))$:

$$H_2 = \frac{1}{n(n-1)} \sum_{i\in\{1,\ldots,n\},j\in\{1,\ldots,n\},i\neq j} H(e_i,e_j) \tag{1.2}$$

1.3.8 Protocole de calcul des mesures H_1 et H_2

Pour calculer les $H(e_i,a_k)$ et les $H(e_i,e_j)$, il nous faut déterminer les fréquences de transitions $p_{i,k,j}$ pour chaque i et chaque k. Ces valeurs sont nécessaires au calcul de H_1 et de H_2.

L'algorithme 1.1 permet d'estimer les $p_{i,k,j}$, pour en déduire les valeurs de H_1 et de H_2. La variable $T_{i,k,j}$ désigne le nombre de transitions entre l'état $e_{i,k}$ et l'état e_j. On aurait pu estimer les valeurs de H_1 et de H_2 dans le cadre de l'apprentissage. Cependant, on aurait couru le risque de mal estimer certaines valeurs, puisque l'objectif de l'apprentissage va être de privilégier certains couples état/action.

1.3.9 Modélisation du flux d'erreurs dû au contexte d'apprentissage

Dans la suite de ce chapitre, nous effectuerons **l'hypothèse (H)** suivante :

> *Le contexte d'apprentissage engendre un flux d'erreurs qui apparaît lors de l'apprentissage et qui n'est pas dû à l'algorithme d'AR lui-même.*

On considère un système utilisant une politique de commande apprise grâce à un algorithme d'AR dans un problème de viabilité [1]. Nous modélisons l'état de fonctionnement de ce système par une chaîne de Markov possédant un état « viable », qui correspond à un état de fonctionnement normal (le système reste dans sa zone de viabilité) et un état « terminal » (figure 1.5). On suppose de plus que l'initialisation du système peut placer celui-ci dans des états viables de nature différente, connectés chacun à un état terminal particulier (voir la figure 1.6).

La production d'un événement fâcheux par le système (sortie de la zone de viabilité) est conditionnée par les valeurs ϵ_j associées à chacune des sources de non-fonctionnement, ainsi que

1. C'est le cas du problème du pendule inversé, que nous traitons dans la section suivante.

Algorithme 1.1 Algorithme de calcul de H_1 et de H_2

Pour chaque état initial e_i, chaque action a_k et chaque état d'arrivée e_j,
  $T_{i,k,j} := 0$
Finpour
Répéter N fois
  initialisation aléatoire des données d'entrée du système dans sa zone de viabilité
  récupérer l'état courant e_i du système
  Pour chaque action a_k, faire
    Répéter
      exécuter a_k
      récupérer l'état courant e_j du système
    Jusqu'à $e_j \neq e_i$
    $T_{i,k,j} := T_{i,k,j} + 1$
  FinPour
FinRépéter
$H_1 := 0$, $H_2 := 0$
Pour chaque état e_i, chaque action a_k et chaque état e_j,
  $p^1_{i,k,j} := \frac{T_{i,k,j}}{\sum_{j=1}^{n} T_{i,k,j}}$
  $H_1 := H_1$ - $p^1_{i,k,j}.\log(p^1_{i,k,j})$
  $p^2_{i,k,j} := \frac{T_{i,k,j}}{\sum_{k=1}^{q} T_{i,k,j}}$
  $H_2 := H_2$ - $p^2_{i,k,j}.\log(p^2_{i,k,j})$
FinPour

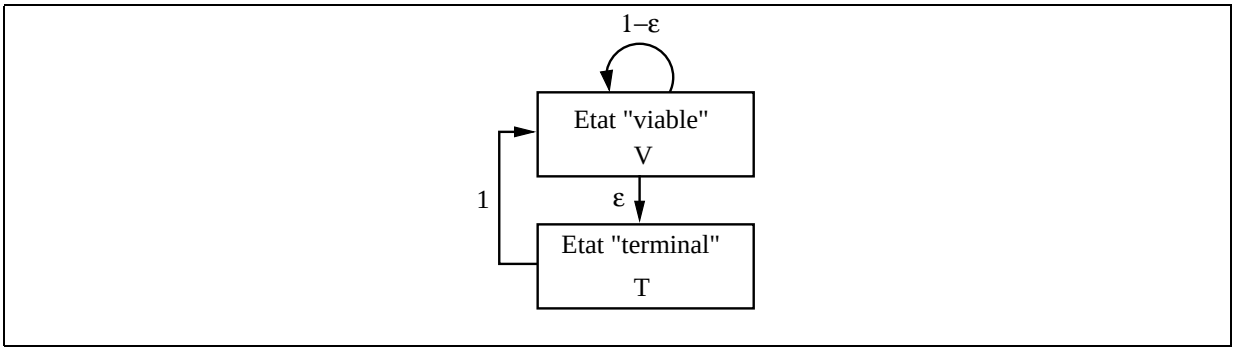


Fig. 1.5 – Hypothèse de modélisation du flux d'événements « non-viable »

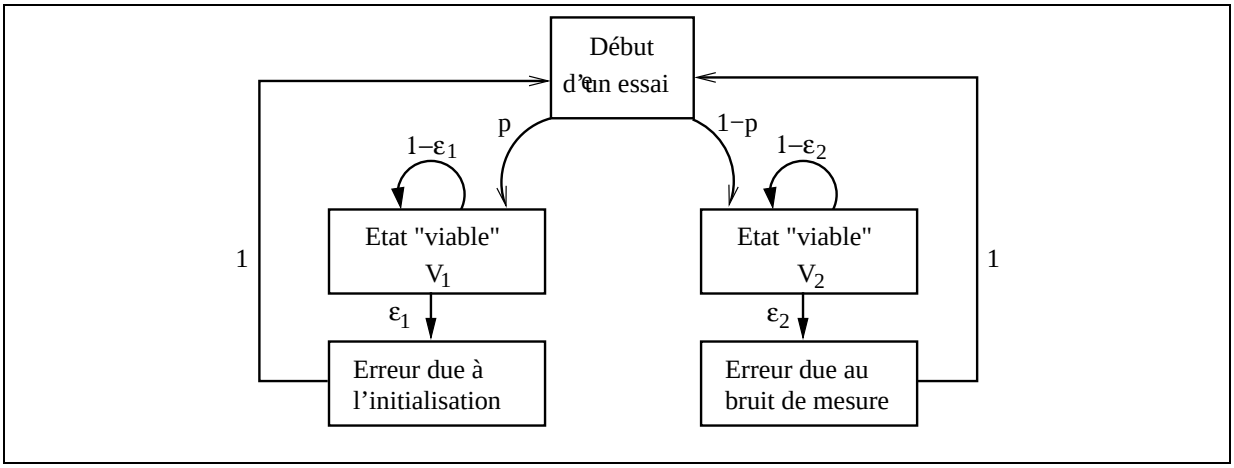


FIG. 1.6 – Modélisation d'un flux d'erreurs possédant deux sources distinctes

la probabilité p d'arrivée dans un état de fonctionnement particulier. Nous allons déterminer l'expression de la durée moyenne avant la première détection d'erreur (durée de viabilité moyenne), ainsi que l'écart-type de cette durée (voir le deux paragraphes qui suivent).

1.3.10 Modélisation d'un flux d'erreurs mono-causal

Nous allons nous intéresser à la modélisation d'un flux d'erreurs lorsqu'il existe une seule cause d'erreur, paramétrée par sa fréquence d'occurrence ϵ.

Considérons un processus dynamique dont l'occurrence d'états non-viables peut être modélisée par une chaîne de Markov à deux états V (pour « viable ») et T (pour « terminal ») : voir la figure 1.5. Il existe trois transitions : l'une de V vers V, possédant une probabilité de franchissement de $1-\epsilon$, avec $\epsilon \in]0,1[$, une autre de V vers T, possédant une probabilité de franchissement de ϵ, puis une troisième de T vers V, possédant une probabilité de transition 1. Cette dernière simule la réinitialisation du système après l'état terminal.

À l'instant initial, le processus se trouve dans l'état V. À chaque pas de temps k, le passage de l'état V à l'état T est formalisé par une variable aléatoire discrète X_k à valeur dans 0,1, suivant une loi de Bernoulli de paramètre ϵ. La valeur de la réalisation x_k de X_k possède la signification suivante : si x_k vaut 0, le processus reste dans l'état V à l'instant k+1, sinon il passe dans l'état terminal T et, à l'instant k+1, il se retrouve dans l'état V.

Nous allons nous intéresser au nombre de pas de temps durant lesquels le processus reste dans l'état V sans passer par l'état T. Pour cela, considérons la variable aléatoire discrète $S_{n,n\in\mathbb{N}^*}$, définie à partir des $X_{k,k\leq n}$ de la manière suivante :

$$S_n = \sum_{k=0}^{n} X_k$$

La variable aléatoire S_n est directement liée à ce nombre de pas de temps, puisque la réalisation s_n vaut 0 uniquement dans le cas où le processus est resté dans l'état V pendant n pas de temps. L'espérance $E[S_n/n]$ correspond au nombre moyen d'erreurs par unité de temps.

Deux points vont nous intéresser :

1. on considère que la valeur de ϵ est connue : comment obtenir la loi de la durée (nombre de pas de temps) séparant deux passages dans l'état T ?
2. on considère que la valeur de ϵ est inconnue : comment estimer la valeur de ϵ grâce à une réalisation s_n de S_n ?

Considérons le point 1. Nous allons faire l'hypothèse que les X_k sont indépendantes. Il est connu que S_n suit une loi binomiale $B(n,\epsilon)$:

$$P(S_n = k) = C_n^k \epsilon^k (1-\epsilon)^{n-k}$$

Le nombre de pas de temps entre l'instant initial et l'instant du premier passage dans l'état T est une variable aléatoire N vérifiant :

$$P(N = k) = P(S_k = 0, X_{k+1} = 1)$$

Les deux événements étant indépendants, il vient :

$$P(N = k) = P(S_k = 0).P(X_{k+1} = 1) = \epsilon(1-\epsilon)^k \tag{1.3}$$

La probabilité pour que le nombre de pas de temps consécutifs dans l'état V soit inférieure à n est donc :

$$P(N \leq n) = \sum_{k=0}^{n} P(N = k) = 1 - (1-\epsilon)^{n+1} \tag{1.4}$$

Considérons à présent le point 2. Calculons en premier lieu l'espérance et la variance de N grâce à l'équation 1.3. Il vient (voir l'annexe A.3) :

$$E[N] = 1/\epsilon$$

Et

$$Var[N] = \frac{1-\epsilon}{\epsilon^2}$$

Lorsque ϵ est suffisamment petit, l'écart-type de N est presque égal à $1/\epsilon$, c'est-à-dire E[N].

Si on considère un p-échantillon $n_1, n_2, ..., n_p$ de N, grâce à la méthode du maximum de vraisemblance, on peut donner un estimateur $\hat{\epsilon}$ (voir l'annexe A.4) :

$$\hat{\epsilon} = \frac{1}{\frac{1}{p}\sum_{k=1}^{p} n_k \ + \ 1} \tag{1.5}$$

L'échantillon nous permettra d'établir un histogramme de la fonction de répartition réelle des durées de viabilité pour le comparer graphiquement avec la fonction de répartition théorique utilisant l'estimation de ϵ comme paramètre. Un exemple de fonction de répartition de N est donnée par la figure 1.7, pour $\epsilon = 10^{-5}$.

Voici à présent quelques exemples numériques. Admettons que le flux d'événements « non-viables », issus d'un processus dynamique en temps discret, soit modélisable en utilisant la variable aléatoire S, avec un paramètre ϵ et que le pas de temps vaille 0.02 s. Le nombre moyen d'erreurs est $E[S_n] = n\epsilon$. La valeur maximum de ϵ pour qu'il y ait en moyenne une erreur par an est : $\epsilon \simeq 6.3410^{-10}$.

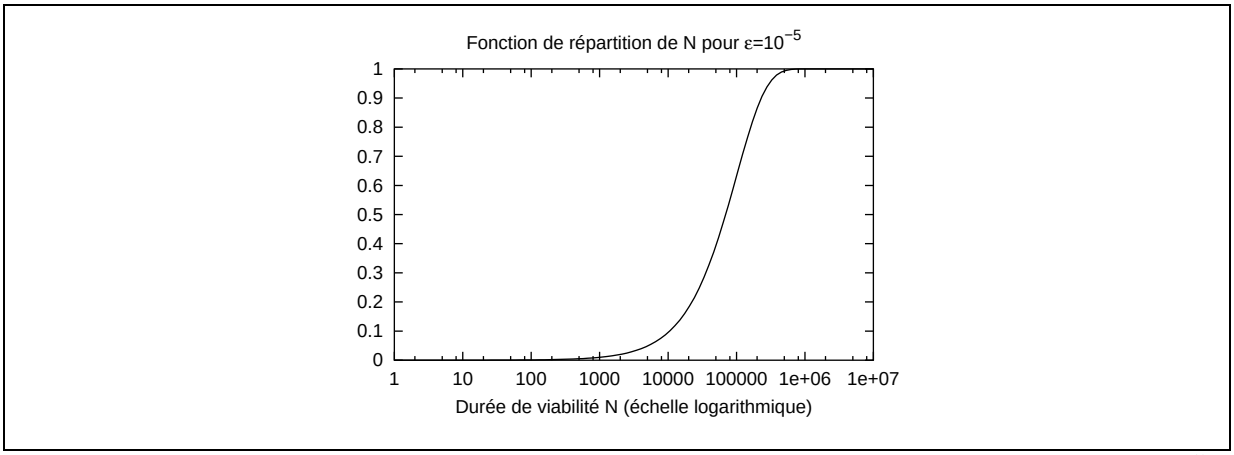


FIG. 1.7 – Exemple de fonction de répartition des durées de viabilité, obtenu pour $\epsilon = 10^{-5}$

Si, à présent, on constate qu'un processus n'est jamais passé par l'état terminal pendant n pas de temps, on peut obtenir une majoration du nombre moyen d'erreurs par jour, avec une certaine confiance. Pour une confiance de 0.99, il suffit de déterminer n pour obtenir P(S_n=0) = 0.99 . Pour $n = 10^5$ (près de 3 minutes, avec un pas de temps de 0.02 s), on obtient $\epsilon < 10^{-6}$. Grâce à l'expression $E[S_n]$, on en déduit que la majoration du nombre d'erreurs moyen par jour est d'environ 4. Si on voulait garantir que cette majoration soit d'une erreur par an, il aurait fallu continuer à observer le processus pendant un peu plus de trois jours.

1.3.11 Modélisation d'un flux d'erreurs bi-causal dépendant de l'état initial du système

Admettons maintenant qu'il existe deux flux d'erreurs, alimentés chacun par un type d'erreur : E_1 ou E_2, caractérisés respectivement par les fréquences d'occurrence d'erreurs ϵ_1 et ϵ_2. On suppose que l'initialisation du système dynamique décide du type de flux d'erreur, suivant une probabilité p. La figure 1.5 montre la chaîne de Markov associée à cette modélisation.

À partir des résultats de la sous-section précédente (équation 1.3), on en déduit la loi de N :

$$P(N = n) = p\epsilon_1(1 - \epsilon_1)^n + (1 - p)\epsilon_2(1 - \epsilon_2)^n$$

L'espérance et la variance de N peuvent être facilement déduites des relations trouvées dans la sous-section précédente :

$$E[N] = \frac{p}{\epsilon_1} + \frac{1-p}{\epsilon_2} Var[N] = p\frac{p + 2 - \epsilon_1}{{\epsilon_1}^2} + (1 - p)\frac{3 - p - \epsilon_2}{{\epsilon_2}^2} + \frac{p(1-p)}{\epsilon_1\epsilon_2}$$

L'estimation des paramètres ϵ_1, ϵ_2 et p est délicate dans le cas général. Par contre, il existe des cas simples. Ainsi, si ϵ_1 et ϵ_2 possèdent des valeurs très éloignées (par exemple, ϵ_1 est bien plus petit que ϵ_2, la fonction de répartition des durées de viabilité est séparable en deux zones bien distinctes : la figure 1.8 montre le cas où $\epsilon_1 = 10^{-5}$, $\epsilon_2 = 10^{-2}$ et p=0.5 . On peut affirmer avec une grande certitude que si le système passe par un état terminal au bout de moins de 1000 pas de temps, il est très probable que l'erreur soit E_2. Au contraire, si le temps de viabilité est supérieur à 1000, il est presque certain que l'erreur soit E_1. La connaissance de ces zones d'influence peut donc permettre d'estimer les valeurs de ϵ_1 et de ϵ_2, en ne tenant compte que des durées de viabilité comprises dans leur zone d'influence respective.

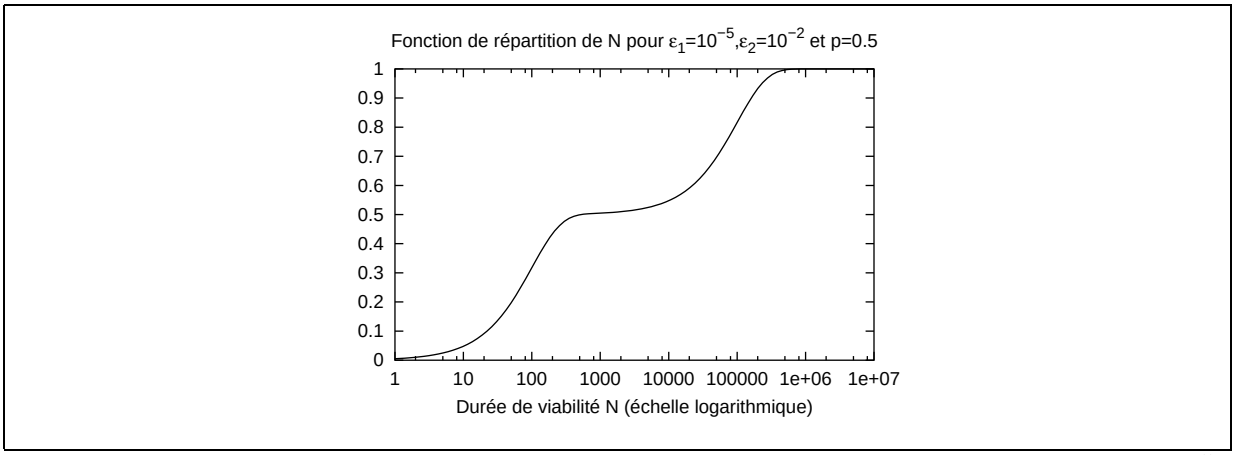


FIG. 1.8 – Exemple de densité de répartition des durées de viabilité, obtenu pour $\epsilon_1 = 10^{-5}$, $\epsilon_2 = 10^{-2}$ et p=0.5

1.4 Expérimentations autour du problème du pendule inversé

1.4.1 Objectif

Le problème du pendule inversé fait partie des bancs d'essais les plus utilisés en AR. L'article de référence de Barto et Sutton l'utilise et beaucoup de techniques d'AR ont été testées grâce à celui-ci (voir la présentation du problème en annexe A.2.1).

Une caractéristique de ce problème est que l'emploi de deux ou trois mauvaises commandes consécutives peut suffire à faire sortir le système de sa zone de viabilité. Or, on sait que l'AR en lui-même doit permettre d'établir une politique de commande correcte dans le sens où si l'état courant est toujours parfaitement identifié et si la topologie des état est bien formée, alors on saura appliquer la bonne commande après apprentissage[1].

Nous avons donné les sources d'incertitude sur la qualité de la politique de commande trouvée après l'apprentissage. Celles-ci sont difficilement maîtrisables et rendent l'interprétation des résultats d'apprentissage délicate : on sent intuitivement que cela peut rendre le système non fiable et le résultat non prédictible, mais il est difficile de trouver la cause d'un résultat d'apprentissage (qu'il soit bon ou mauvais).

Pour un problème de viabilité, l'expérimentateur peut choisir comme critère de réussite de l'apprentissage un nombre arbitraire d'itérations consécutives sans échec. Il est clair que si le problème respecte l'hypothèse (H) que nous avons introduite dans la sous-section 1.3.9, les performances de l'algorithme d'AR déduites de ce critère dépendent de la valeur de ce nombre limite. La qualité des résultats d'apprentissage n'a donc qu'une valeur relative.

La sous-section suivante présente les résultats d'un des rares travaux en AR concernant l'influence d'une dégradation de la qualité des données sur la qualité de l'apprentissage et sur la durée d'apprentissage. Ils constituent une base pour notre analyse et mettent en évidence des interrogations.

1. Cela est traduit par l'hypothèse markovienne concernant le problème de décision de l'action à entreprendre à tout instant.

Comment peut-on interpréter un résultat d'AR? Quelles sont les causes de l'échec ou de la réussite d'un apprentissage? Comment peut-on introduire la notion de fiabilité d'une politique de commande issue de l'AR? L'objectif de cette section est de donner des éléments de réponse à ces questions autour des problématiques de fiabilité et de prédictibilité. Pour cela, nous utiliserons les outils de mesure définis dans la section précédente.

1.4.2 Analyse préliminaire des résultats antérieurs sur l'influence du bruit de mesure sur l'apprentissage par AR

Peu de travaux existent, qualifiant précisément l'influence du bruit de mesure sur l'AR. Nous choisissons de mentionner l'étude de Pendrith (voir [Pendrith, 1994]). Comme le montrent d'autres de ses publications ([Pendrith et McGarity, 1998], [Pendrith, 1999]), les efforts de Pendrith se sont centrés sur la réduction des effets néfastes des bruits de mesures sur l'apprentissage, dans des conditions réelles. Son rapport interne de 1994 est une base de travail intéressante pour nous, car Pendrith y a étudié minutieusement l'influence du bruit de mesure sur les algorithmes AHC et Q-Learning, en comparaison avec ses propres algorithmes P-Trace et Q-Trace. Le banc d'essai était le pendule inversé (en simulation), avec le découpage indiqué dans [Barto et al., 1983].

Voici les conditions expérimentales. Pendrith définit le temps d'apprentissage comme le nombre moyen d'essais nécessaires pour aboutir au premier succès de la politique de commande apprise. Il définit ce succès en fonction du nombre d'itérations consécutives pour lesquelles le système reste viable, et fixe ce nombre à 10.000 .

Au début de chaque essai, l'état du système est choisi au hasard dans une zone de l'espace d'états[1]. L'algorithme de choix de la commande utilise la loi de probabilité de Boltzmann (voir [Lin, 1992]), avec un paramètre de température T choisi au début de chaque essai dans l'intervalle]0,0.1].

Lorsqu'un essai est réussi, Pendrith fixe la politique de commande du système[2] et la teste à partir de 20 positions initiales différentes du chariot et du pendule. Sur ces 20 tests, il compte le nombre de tests ayant aboutit à un succès (c'est-à-dire 10.000 itérations consécutives sans échec), ce qui qualifie le « score » de l'apprentissage avec une note de 0 à 20.

Pendrith introduit des bruits de mesure suivant une loi uniforme. Ces bruits sont de deux types :

- type 1 : l'amplitude maximum du bruit est proportionnelle à la valeur de la variable (donc, peu de bruit pour un système proche de l'équilibre : un biais très faible est rajouté pour que le bruit ne soit pas nul à l'équilibre)
- type 2 : l'amplitude maximum du bruit est constante, quelle que soit la valeur de la variable considérée

Il spécifie les amplitudes de bruit, allant de 0% (pas de bruit) à 8% (amplitude maximum pour les tests de Pendrith).

1. Celle-ci est la même que pour l'expérience menée par Barto et Sutton. Nous la préciserons dans notre protocole expérimental. 2. Pour chaque état, il choisit la commande associée à la qualité la plus grande (politique gloutonne).

TAB. 1.1 – Résultats obtenus par Pendrith pour les méthodes AHC et Q-Learning (bruit de type 1)

Bruit (%)	Score AHC	Score Q-L	NME AHC (ENE AHC)	NME Q-L (ENE Q-L)
0	9.6	10.1	56.3 (20.6)	403.6 (184.3)
1	9.7	8.1	61.5 (36.4)	503.0 (267.3)
2	8.1	11.9	57.9 (16.3)	690.9 (504.6)
3	7.0	10.1	137.2 (252.0)	1080.6 (688.9)
4	4.9	6.8	100.9 (79.7)	1905.0 (1484.3)
5	4.7	6.3	473.3 (857.5)	7949.3 (8576.8)
6	2.8	2.0	125.5 (111.4)	22937.5 (18037.4)
7	1.6	1.0	58001.6 (251354.0)	97301.6 (95936.6)
8	0.4	0.4	1900.9 (5970.7)	867567.4 (845976.0)

NME représente le Nombre Moyen d'Essais avant la réussite de l'apprentissage
ENE représente l'Écart-type du Nombre d'Essais avant la réussite de l'apprentissage
Q-L est l'abréviation de Q-Learning

Les résultats que Pendrith obtient pour les méthodes AHC et Q-Learning sont résumés dans le tableau 1.1. Nous n'indiquons que ceux obtenus pour les bruits de type 1. Des résultats analogues sont trouvés pour les bruits de type 2. Ils font apparaître les éléments suivants :

- le score sans bruit est loin d'être maximum
- les algorithmes AHC et Q-Learning sont très sensibles aux bruits de mesure, ce qui se traduit par une augmentation très nette du nombre d'essais avant que l'apprentissage soit réussi et une diminution du score d'apprentissage
- la variance du nombre d'essais nécessaires à la réussite de l'apprentissage est importante

Pendrith note également que la plupart des essais sont réussis.

Nous pouvons faire quelques remarques sur ces résultats. En premier lieu, le fait que la majorité des apprentissages se terminent par un succès n'est pas surprenant. Nous avons obtenu des résultats similaires en choisissant au hasard, pour chaque état du système (voir la description du problème du pendule inversé), la commande à appliquer, puis en testant cette politique de commande. Il ressort de cette expérience que le fait de trouver au hasard une politique de commande satisfaisant le critère de réussite de Pendrith n'est pas un événement rare (il faut effectuer quelques millions d'essais pour cela[1]). Quels scores Pendrith aurait-il obtenu si le critère de réussite avait été 100.000 itérations consécutives ou davantage ? Il n'est pas garanti que le résultat aurait été similaire à celui présenté dans la table 1.1.

En second lieu, on remarque que les scores d'apprentissage sans bruit de mesure sont médiocres (à peine la moitié des tests s'achèvent par un succès). On peut expliquer ce mauvais résultat par un manque d'exploration de certains états dans lesquels le système est initialisé au début de chaque essai. Cette explication est plausible, mais on peut en trouver d'autres :

- la zone d'initialisation du système au début de chaque essai comporte des régions à partir desquelles le système ne peut pas rester viable très longtemps.
- le mécanisme de choix d'action, utilisation une loi de probabilité de Boltzmann, possède un paramètre température T que Pendrith choisit au début de chaque essai dans l'intervalle]0,0.1]. Ce paramètre permet de choisir une commande possédant une qualité non

1. Ce résultat est étonnant au premier abord. En effet, dans cette expérience, le système comporte 162 états et 2 commandes possibles. Le nombre de combinaisons possibles est donc de $2^{162} \simeq 5.810^{48}$. Le fait de trouver une politique de commande correcte aussi rapidement conduit à penser que le nombres d'états vraiment utiles est beaucoup plus faible que 162.

maximum, ce qui induit une exploration plus complète des états du système et évite que l'apprentissage tombe dans un « minimum local ». Lorsque les qualités associées à chaque commande sont très proches et que la température est élevée, l'algorithme de choix de commande effectue en quelque sorte un tirage aléatoire des commandes. Or, il apparaît que, pour le problème du pendule inversé, ce choix aléatoire au niveau des zones de l'espace d'états proche du point d'équilibre est favorable au maintien du système en équilibre. Dans la phase de tests, une seule commande par état est choisie (celle qui est associée à la meilleure qualité), ce qui est forcément moins favorable (la vitesse angulaire du pendule ou la vitesse du chariot peuvent être beaucoup plus importantes que prévu à la sortie de l'état lorsqu'une commande unique y est exécutée.).

Enfin, l'écart-type associé au nombre d'essais nécessaires avant la réussite de l'apprentissage est de l'ordre de grandeur du nombre moyen d'essais. Il y a donc une grande variabilité des paramètres fondamentaux de l'apprentissage : le temps d'apprentissage avant l'obtention d'un succès et le score d'apprentissage. Nos interrogations sont destinées à sensibiliser le lecteur sur le fait que le résultat d'un apprentissage d'AR n'est que peu prédictible, et que l'absence de fiabilité de la politique de commande obtenue après apprentissage est difficilement explicable. Les causes de cette non fiabilité et de cette non prédictibilité proviennent du contexte de l'apprentissage : algorithme de choix de la commande, zone d'initialisation du système, critère d'arrêt de l'apprentissage.

L'objectif de notre exemple applicatif est de répondre à certaines des questions que nous avons posées, sans avoir trouvé de réponse dans le travail de Pendrith :

- l'AR peut-il engendrer une politique de commande fiable, dans le cas du pendule inversé?
- Peut-on préciser les causes d'une non fiabilité?
- Peut-on prévoir, avant l'apprentissage, la nature de son résultat?

Pour répondre à ces questions, nous utiliserons les outils que nous avons présentés dans la section précédente.

1.4.3 Protocole expérimental

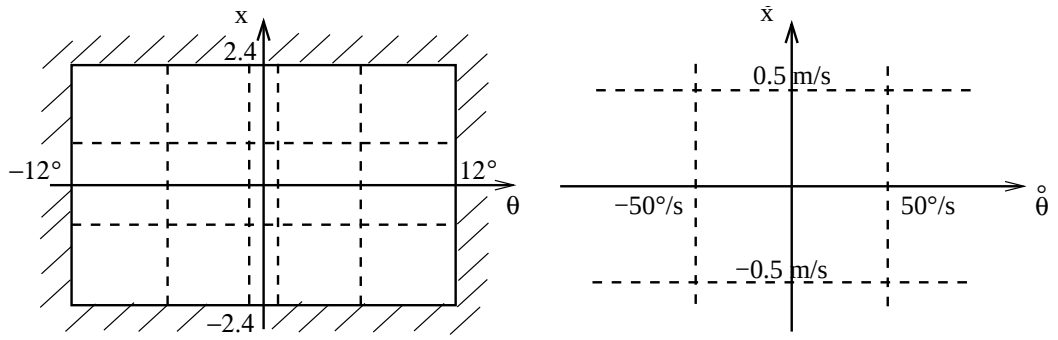
Nous utiliserons la modélisation du flux d'erreurs dû au contexte d'apprentissage, induit par l'hypothèse « (H) » (voir la sous-section 1.3.9).

Pour répondre à la problématique de fiabilité posée dans la sous-section précédente, nous sommes amené à reconsidérer la durée limite de viabilité (fixée arbitrairement) pour laquelle on suppose qu'un essai est réussi. Nous l'avons fixée à **100 millions d'itérations**. Si (H) est respectée, l'observation d'un essai réussi permet de majorer ϵ avec une confiance de 0.99 :

$$\epsilon \leq 10^{-9}$$

Grâce à l'expression $E[S_n]$, on en déduit une majoration du nombre moyen d'erreurs par an de 1.5 .

Dans cette série d'expériences, nous reprenons la topologie des états constitués dans [Barto et al., 1983], qui découpe l'espace d'états généré par les quatre variables d'état x,$\dot{x}$,θ,$\dot{\theta}$ en 162 boîtes (voir la figure 1.9 pour les coupes selon les axes θ,x puis selon les axes $\dot{\theta}$,$\dot{x}$).



La zone hachurée du graphique de gauche correspond à une non-viabilité du système. Les traits en pointillés marquent les zones frontières entre les boîtes (pour la figure de gauche, les axes sont des frontières).

FIG. 1.9 – Coupes du découpage de l'espace d'état pour le problème du pendule inversé

Pour chaque essai, l'initialisation du système s'effectue en choisissant aléatoirement le quadruplet $(x,\dot{x},\theta,\dot{\theta})$ dans l'hypercube [-0.8,0.8]×[-0.5,0.5]×[-6°,6°]×[-0.87,0.87]. Ce domaine est choisi classiquement en AR pour le problème du pendule inversé. Un essai est stoppé lorsque le système est resté viable pendant 100 millions d'itérations consécutives ou lorsque le système est sorti de sa zone de viabilité.

Un apprentissage comporte une série de 2000 essais, sauf mention contraire.

La technique d'apprentissage choisie est le Q-Learning avec trace d'éligibilité (voir les algorithmes en annexe). La méthode AHC n'est pas considérée dans le corps de ce document, car elle est reconnue comme étant instable dans le cas du pendule inversé ([Watkins, 1989], [Pendrith, 1994]), même lorsque les données d'entrée sont parfaites.

Nous avons effectué plusieurs essais préliminaires avant de déterminer la politique de choix d'action. Nous avons retenu une méthode pseudo-exhaustive : on a une probabilité P de choisir la meilleure action (P décroît suivant le temps d'une manière linéaire) et 1 - P de choisir l'autre. Nous avons choisi cette méthode car elle semble donner de meilleurs résultats que la méthode basée sur la distribution de Boltzmann.

Les mesures de H_1 et H_2 sont effectuées suivant le protocole décrit par l'algorithme 1.1 [1] (paragraphe 1.3.8, page 16).

Dans les expériences présentées ci-dessous, on teste différents modèles de bruit de mesure, les autres paramètres restants inchangés.

- données d'entrée parfaites. La connaissance des états du système est absolue
- données d'entrée artificiellement bruitées avec un bruit gaussien, d'amplitude σ, **appliqué uniquement sur l'entrée** θ
- données d'entrée possédant un certain taux τ_{VA} de valeurs aberrantes (appliqué sur toutes les variables d'entrée)
- introduction d'un état choisi aléatoirement, avec un taux τ_{EA}

1. Le nombre d'itérations N est fixé à 100 millions.

1.4.4 Analyse des mesures H_1 et H_2 lorsque la qualité des données d'entrée est dégradée

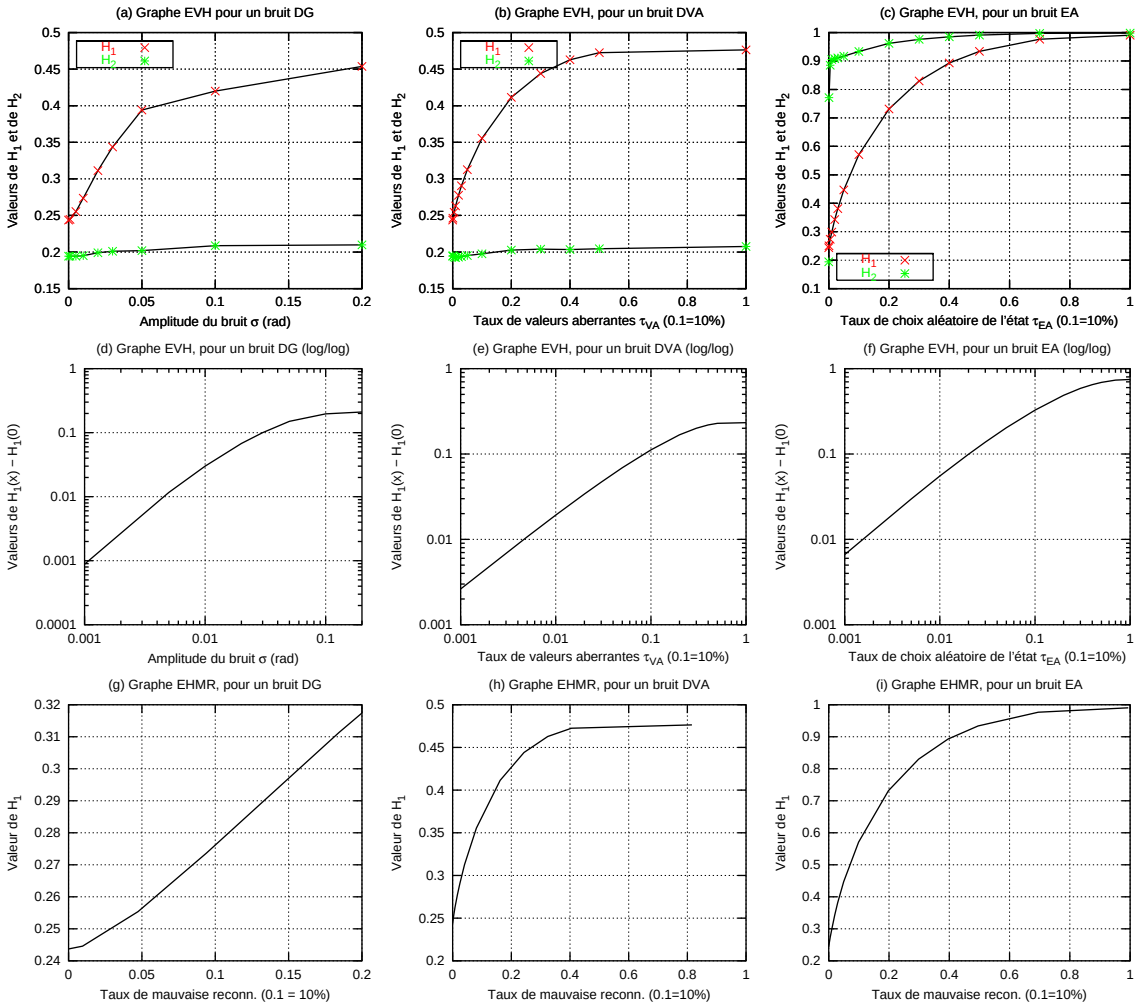
Avant d'entamer l'étude des résultats d'apprentissage, étudions le comportement des mesures H_1 et H_2 lorsque les paramètres du contexte d'apprentissage changent : la qualité de l'entrée θ (bruit gaussien d'amplitude σ variable, taux variable de données aberrantes) ou la nature de l'état courant du système (l'état du système peut être choisi aléatoirement avec un taux variable) [1]. Si H_1 et H_2 sont capables de mesurer la qualité du contexte d'apprentissage, il faut vérifier si les conditions suivantes sont remplies :

1. H_1 et H_2 doivent être deux fonctions strictement croissantes de l'amplitude du bruit de mesure
2. les variations de H_1 et de H_2 doivent être suffisamment importantes

Les graphes (a),(b) et (c) de la figure 1.10 montrent précisément l'évolution de H_1 et de H_2 en fonction de l'amplitude du bruit de mesure, pour chacune des catégories de bruit. Les calculs de H_1 et de H_2 sont obtenus en utilisant l'algorithme 1.1, page 17, avec une valeur du paramètre N égale à 100 millions d'itérations. Nous remarquons que la première condition est remplie par H_1 et H_2, mais que les variations de cette dernière ne sont pas assez importantes pour les graphes (a) et (b) (bruit gaussien et introduction de valeurs aberrantes de θ). Dans la suite de cette sous-section, nous donnerons une explication à cette faible variation. Mais, pour l'instant, nous allons nous intéresser plus précisément à H_1, qui répond à nos deux conditions. Les graphes (d), (e) et (f) de la figure 1.10 précisent l'évolution de H_1 lorsque l'amplitude du bruit de mesure est faible (échelle log/log). On constate une relation linéaire, à l'échelle log/log, entre l'amplitude du bruit et H_1, ce qui signifie que H_1 peut être mise sous la forme $b.\tau^a$, τ représentant l'amplitude du bruit de mesure et « a » la pente de la droite. Cette pente a dépend de la nature du bruit : elle vaut environ 0.9 si le bruit est gaussien (assimilation linéaire valide pour une amplitude inférieure à 0.02), environ 2.6 s'il existe un taux de mesures aberrantes de θ (assimilation linéaire valide pour une amplitude inférieure à 0.1) et environ 12.4 suivant le taux de choix aléatoire de l'état courant (assimilation linéaire valide pour une amplitude inférieure à 0.05). H_1 permet donc de discriminer les trois catégories de bruit de mesure. Ainsi, des pentes à l'origine différentes indiquent des amplitudes de dégradation du contexte d'apprentissage différentes, qui sont cohérentes avec notre intuition : l'augmentation du taux de choix d'un état aléatoire aboutit plus rapidement à un contexte d'apprentissage dégradé que celle du taux de valeurs aberrantes de θ, et plus rapidement encore que celle de l'amplitude σ d'un bruit gaussien. H_1 est donc un bon indicateur de la dégradation du contexte d'apprentissage due à l'introduction d'un bruit de mesure.

Un autre indicateur de la dégradation du contexte d'apprentissage en présence d'un bruit de mesure serait le taux de mauvaise reconnaissance de l'état courant. L'état courant est mal reconnu s'il est différent de l'état théorique, obtenu avec des données non-bruitées. Les graphes (g), (h) et (i) de la figure 1.10 montrent le lien fonctionnel qui existe entre ce taux de mauvaise reconnaissance et H_1, pour les trois catégories de bruits de mesure. On remarque que H_1 possède une relation linéaire forte avec le taux de mauvaise reconnaissance, pour des valeurs de ce dernier inférieures à 0.55 (graphe (g)), alors que les deux autres graphes montrent une convergence exponentielle vers le taux maximal de mauvaise reconnaissance, avec une pente à l'origine nettement supérieure à 1. On en déduit que la mesure H_1 est nettement plus sensible que le taux de mauvaise reconnaissance à la détérioration de la qualité des données d'entrée par l'introduction

1. Des tests ont également été réalisés lorsqu'on change les limites du découpage de l'espace d'états. Ils ne sont pas inclus dans ce document de thèse, car ils montrent une faible variation de H_1 et de H_2.



Légende :
« EVH » Évolution de la Valeur de H_1 et de H_2 (abscisses : amplitude du bruit de mesure, taux de valeurs aléatoires $[0.1 = 10\%]$; ordonnées : valeur normalisée de H_1 et/ou de H_2).
« EHMR » Évolution de la valeur de H_1 en fonction du taux de Mauvaise Reconnaissance de l'état courant du système.
« DG » Données issues de θ bruitées avec un bruit gaussien, d'amplitude σ.
« DVA » Données issues de θ possédant un taux de valeurs aberrantes τ_{VA}.
« EA » L'état du système peut être choisi aléatoirement, avec une fréquence d'apparition τ_{EA}.
Commentaires :
Les graphes (a),(b) et (c) montrent l'évolution des valeurs de H_1 et H_2 lorsque l'amplitude du bruit de mesure augmente, suivant les trois types de bruits. Les graphes (d), (e) et (f) sont construits respectivement à partir de (a), (b) et (c), en considérant une échelle log/log. L'intérêt est de constater que lorsque l'amplitude du bruit est faible, H_1 peut se mettre sous la forme $b.\tau^a$, avec τ l'amplitude du bruit de mesure, et a et b les paramètres des droites constatées dans les graphes (d),(e) et (f) lorsque τ est suffisamment faible. Enfin, les graphes (g), (h) et (i) montrent qu'il existe une relation fonctionnelle entre le taux de mauvaise reconnaissance de l'état courant du système et la valeur de H_1. On s'aperçoit que s'il existe une corrélation quasi-linéaire pour un bruit « DG », pour les deux autres bruits, H_1 augmente beaucoup plus vite que le taux de mauvaise reconnaissance.

Fig. 1.10 – Graphes associés à la sous-section 1.4.4

de mesures aberrantes. Par contre, H_1 est équivalente à la mesure du taux de reconnaissance lorsqu'on introduit un bruit gaussien.

L'avantage de H_1 par rapport au taux de mauvaise reconnaissance est donc résumé par les deux points suivants :

- H_1 est plus sensible que le taux de mauvaise reconnaissance pour des faibles amplitudes du bruit de mesure
- H_1 permet de mesurer la dégradation du contexte d'apprentissage qui provient d'autres sources que l'introduction d'un bruit de mesure : la valeur de H_1 est non nulle lorsque les données d'entrée sont parfaites (les graphes (a), (b) et (c) donnent une valeur à l'origine de 0.24), car elle dépend également de la topologie des états du système.

Revenons à la différence de comportement de H_1 et de H_2, constatée dans les graphes (a) et (b). Que signifie-t-elle ? Rappelons que H_2 mesure la capacité à déterminer l'action qui a permis le passage d'un état e_i à un état e_j, connaissant e_i et e_j. La faible variation de H_2, même pour des amplitudes très importantes du bruit de mesure, signifie que cette capacité n'est, curieusement, pas beaucoup dégradée. La différence essentielle entre les bruits des graphes (a) et (b) d'une part, et celui de (c) d'autre part, est que les deux premiers impliquent uniquement l'entrée θ. Cela signifie que, dans ces deux cas, si on prend deux états successifs atteints par le système, une mauvaise reconnaissance de l'un ou l'autre des états, en raison du bruit introduit sur θ, ne change pas beaucoup la qualité de la prédiction sur l'action qui a provoqué ce changement : on en déduit que la variable θ est peu discriminante pour cette prédiction. Ce résultat s'explique par les caractéristiques du problème du pendule inversé : la variable amenant le plus fréquemment un changement d'état est $\dot{\theta}$, qui devient positive ou négative en très peu d'itérations, suivant la direction de la force qui est appliquée au chariot (cela est dû à la faible inertie de la tige). H_2 est donc un indicateur utile, puisqu'il montre que la variable θ est moins discriminante que ce qu'on pourrait penser intuitivement. En particulier, H_2 nous a permis de constater qu'un découpage beaucoup plus grossier de l'axe θ suffirait.

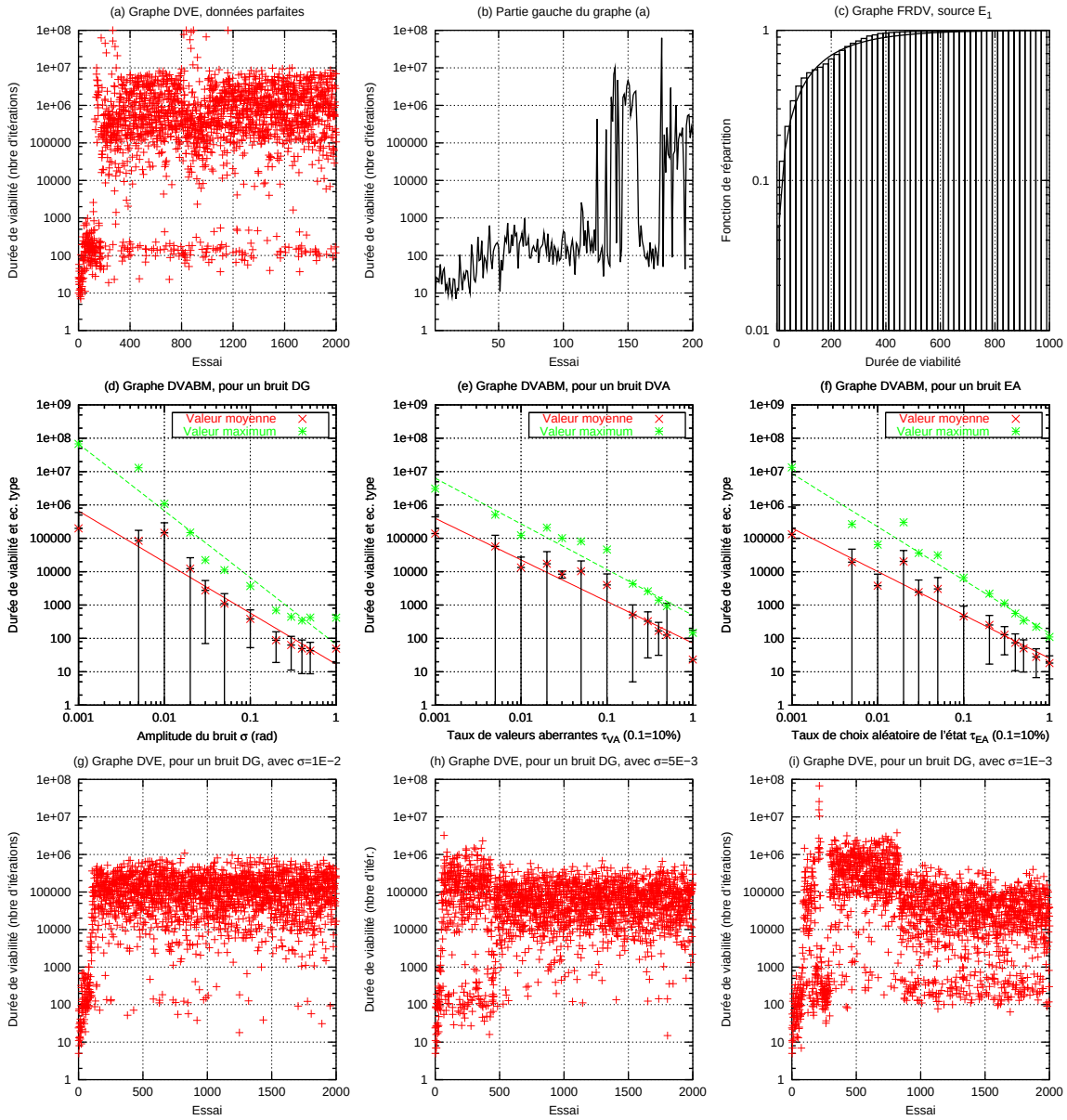
En conclusion, nous venons de montrer l'intérêt de l'utilisation des mesures H_1 et H_2. Dans notre expérience, nous avons montré que H_1 peut effectivement qualifier la dégradation de la qualité du contexte d'apprentissage due à l'introduction d'un bruit de mesure. Des expériences, que nous ne faisons pas figurer dans ce document, montrent une faible influence d'une modification des limites du découpage selon θ sur la valeur de H_1 et de H_2. Elles confirment les résultats, apparemment étonnants, obtenus en étudiant H_2, qui montrent qu'une dégradation des données mesurant θ n'influencent pas beaucoup la valeur de H_2. Nous avons déduit de ces deux résultats que θ est beaucoup moins discriminante que ce qu'on pourrait penser intuitivement.

Un résultat découlant de notre étude est que notre modélisation du problème du pendule inversé ne fournit pas un contexte d'apprentissage respectant la propriété (P_ϵ) : les valeurs de H_1 et de H_2 sont loin d'être proches de 0, même lorsque les données d'entrée sont parfaites.

1.4.5 Découverte des sources d'erreur du contexte de l'apprentissage

Nous allons nous intéresser à la répartition des durées de viabilité au cours des essais, lorsqu'on dégrade les données d'entrée du système. L'hypothèse (H) sera mise à l'épreuve.

Le système, après apprentissage, est-il fiable ? Le graphe (a) de la figure 1.11 montre l'évolution de la durée de viabilité au cours des essais d'apprentissage, lorsque les données d'entrée



Légende :
« DVE » : Durée de Viabilité en fonction du nombre d'Essais d'apprentissage.
« DVABM » : Durée de Viabilité en fonction de l'Amplitude du Bruit de Mesure.
« FRDV » : Fonction de Répartition des Durées de Viabilité.
« DG » Données issues de θ bruitées avec un bruit gaussien, d'amplitude σ.
« DVA » Données issues de θ possédant un taux de valeurs aberrantes τ_{VA}.
« EA » L'état du système peut être choisi aléatoirement, avec une fréquence d'apparition τ_{EA}.
Commantaires :
Les graphes (d),(e) et (f) montrent deux droites, calculées selon le critère des moindres carrés. Chaque droite correspond, dans le graphe log/log, à une modélisation du lien entre la durée de viabilité (moyenne ou maximum) et l'amplitude du bruit de mesure, prenant la forme exp(b).τ^a, avec τ l'amplitude du bruit de mesure, « a » la pente de la droite et « b » sa valeur à l'origine. Pour des bruits de type DB, DVA et EA, on trouve respectivement a_{DG}=-1.53, a_{DVA}=-1.24 et a_{EA}=-1.30.

Fig. 1.11 – Graphes associés à la sous-section 1.4.5

sont parfaites. La première constatation est que l'apprentissage est réussi pour plusieurs essais. D'autre part, la phase d'apprentissage est nettement visible, puisqu'une progression de la durée de viabilité existe jusque vers l'essai 140 (le graphe (b) de la figure 1.11 montre la partie du graphe (a) pour les essais allant de 1 à 200). Mais, nous constatons que, même lorsque les données d'entrée sont parfaites, le système peut sortir de son domaine de viabilité au bout de très peu d'itérations. Ainsi, on distingue clairement, sur le graphe (a), deux bandes horizontales, correspondant à deux zones dans lesquelles les durées de viabilité se situent majoritairement : une zone se situe autour de 150 itérations et une autre est centrée à, environ, un million d'itérations. Donc, une première réponse à la question posée en début de paragraphe est que le système **peut être fiable**, mais que **nous ne savons pas prédire, au début d'un essai, s'il va se terminer par un succès ou non**.

La nature du graphe (a) peut-elle être expliquée en posant l'hypothèse (H) (voir la sous-section 1.3.9) ? Le nombre moyen d'essais se terminant après au moins 100.000 itérations entre deux essais terminés rapidement (moins de 1000 itérations viables) est d'environ 10, avec un écart-type de 20. D'autre part, il y a très peu d'échecs dont le nombre d'itérations est supérieur à 1000 et inférieur à 100.000. L'existence de deux bandes horizontales, qui semblent ne pas évoluer, du moins jusqu'à l'essai 2000, nous montre que la politique de commande paraît avoir convergé. Pour expliquer l'existence de ces deux bandes stationnaires, **il est légitime de poser l'hypothèse (H)**. Dans ce cas, ces deux bandes s'expliquent par la présence de deux sources d'erreurs, dont les fréquences d'apparition sont différentes. L'étude théorique de ce cas est donnée dans la sous-section 1.3.11. La source d'erreurs E_1 caractérise la bande centrée autour de 150 itérations, alors que la source d'erreurs E_2 caractérise celle autour de un million d'itérations. À l'initialisation d'un essai d'apprentissage, la probabilité p de faire face à une erreur provenant de E_1 est égale à 1/10 (d'après les résultats donnés par le graphe (a)). Notre objectif est de déterminer la valeur du paramètre ϵ_1. Pour obtenir un résultat statistiquement valide, nous effectuons **un nouvel apprentissage sur 100.000 essais, mais avec un nombre maximum d'itérations de 1000 par essai** [1], pour récupérer les durées de viabilité issues de la source d'erreurs E_1. Puis, nous établissons l'histogramme de la répartition des durées de viabilité sur ce nouvel ensemble. Nous supposons alors que celui-ci constitue une observation d'un flux unique, de paramètre ϵ_1. La valeur de $\hat{\epsilon_1}$ est donnée par l'équation 1.5, page 19, en connaissant la moyenne des durées de viabilité : nous trouvons $\hat{\epsilon_1} = 5.710^{-3}$. Il nous faut, à présent, vérifier que la répartition réelle des durées de viabilité correspond à la théorie : le graphe (c) de la figure 1.11 montre une superposition satisfaisante de l'histogramme issu des observations avec la fonction de répartition théorique. **L'hypothèse (H) est donc validée** dans le cas du graphe (a), c'est-à-dire lorsque les données d'entrée sont parfaites.

À quoi peut-on relier les sources d'erreurs E_1 et E_2 ? D'après notre modèle (voir la sous-section 1.3.11), c'est au moment de l'initialisation d'un essai d'apprentissage que, suivant une probabilité p, la source d'erreur E_1 est sélectionnée, et suivant une probabilité 1-p, la source E_2 est choisie. Quel type d'erreurs E_1 représente-t-elle ? Deux hypothèses sont envisagées :

1. l'erreur est exclusivement imputable à l'état initial du système, choisi aléatoirement au début de chaque essai. Certains états initiaux conduiraient forcément vers l'extérieur de la zone de viabilité du système, indépendamment de l'algorithme d'apprentissage.

1. Le choix de la limite de 1000 essais est fait en fonction des résultats du graphe (a) : nous avons constaté que les deux sources d'erreurs forment deux bandes clairement séparées, ce qui permet de dire avec une quasi-certitude que si un essai se termine après moins de 1000 itérations, la source d'erreurs associée est E_1.

2. l'erreur est causée par une convergence trop rapide et erronée des valeurs qualités associées à certains états du système. La non rectification des erreurs portant sur ces qualités serait due à un mauvais paramétrage du dilemme exploration/exploitation gérant le choix d'une commande à tout moment.

Examinons la première hypothèse. Dans ce cas, la probabilité p correspond à la probabilité de choisir un état initial aboutissant forcément à un échec rapide. Si cette hypothèse est valide, le paramètre p doit être une constante qu'on peut retrouver pour tous les apprentissages. Or, cela n'est pas le cas. Les graphes (g) et (i) de la figure 1.11 montrent que le nombre de valeurs de la durée de viabilité se situant autour de 150 itérations n'est pas constant (beaucoup plus important pour le graphe (i) que pour le graphe (g)). **La première hypothèse est donc rejetée**.

Prenons, à présent, la seconde hypothèse. Il y aurait un manque d'exploration des états du système, dû à l'algorithme de choix de la commande. Cela est probable, car celui-ci privilégie de plus en plus, au fil des essais, une politique gloutonne (choix de la meilleure qualité). Cela permet de diminuer, au fil des essais, le nombre d'erreurs dans le choix de la commande, mais ralentit la mise à jour des qualités de certains états dont la probabilité d'exploration est rendue trop faible. La nature de l'état initial joue un rôle dans cette hypothèse. En effet, au cours de l'apprentissage, des pseudo-cycles [1] états/action sont formés. L'accès à un pseudo-cycle particulier, donc à un ensemble d'états particuliers, dépend de l'état initial du système. Or, si un pseudo-cycle possède un état dont la qualité associée implique le choix d'une commande responsable d'une sortie du domaine de viabilité, la durée de viabilité dépendra de la probabilité d'atteinte de ce pseudo-cycle. Et celle-ci dépend à la fois de l'évolution de l'apprentissage (une qualité est erronée) et du choix de l'état initial (menant au choix d'une mauvaise commande). La deuxième hypothèse est donc probable. Cependant, elle n'est pas compatible, en théorie, avec l'aspect du graphe (a). En effet, l'algorithme de choix de l'action n'est jamais totalement glouton : il autorise, avec une probabilité non nulle, que la qualité erronée puisse être rectifiée. Or, cette rectification, lorsqu'elle a lieu, peut avoir un impact sur la valeur de p et sur la durée moyenne de viabilité, ce que ne montre pas le graphe (a). Mais, les graphes (g), (h) et (i) font apparaître **plusieurs zones**, après ce que nous avions appelé la zone d'apprentissage (les 140 premiers essais), en plus de la zone d'apprentissage : dans chacune de ces zones, on peut remarquer les deux bandes, mais avec une différence au niveau de la valeur de p. Ainsi, dans le graphe (h), il existe deux zones : l'une comprise environ entre l'essai 100 et l'essai 500, et l'autre débutant après l'essai 500. Dans le graphe (i), on constate trois zones : la première est comprise entre les essais 100 et 250, une autre entre les essais 250 et 800, puis enfin la dernière. **Nous concluons cette analyse en validant l'hypothèse 2**.

Il nous reste à expliquer la cause de la source d'erreurs E_2, de paramètre ϵ_2. En comparant les graphes (a), (g), (h) et (i), utilisant respectivement des valeurs d'entrées parfaites, un bruit gaussien d'amplitude $\sigma = 0.01$, un bruit gaussien d'amplitude $\sigma = 0.005$ et un bruit gaussien d'amplitude $\sigma = 0.0001$, nous observons deux faits :

- les valeurs de ϵ_2 associées aux données des quatre graphes sont différentes.
- pour les graphes (g),(h) et (i), il n'existe pas de valeur unique pour ϵ_2, mais plusieurs valeurs, qui dépendent des zones dont nous venons de parler ci-dessus.

Nous faisons l'hypothèse que E_2 regroupe deux causes d'erreurs principales :

- l'amplitude du bruit de mesure

1. Nous les appelons « pseudo-cycles » car le système ne parcourt jamais un véritable cycle d'états : il parcourt un ensemble d'états pendant un certain temps, le quitte pour en visiter un autre, etc. .

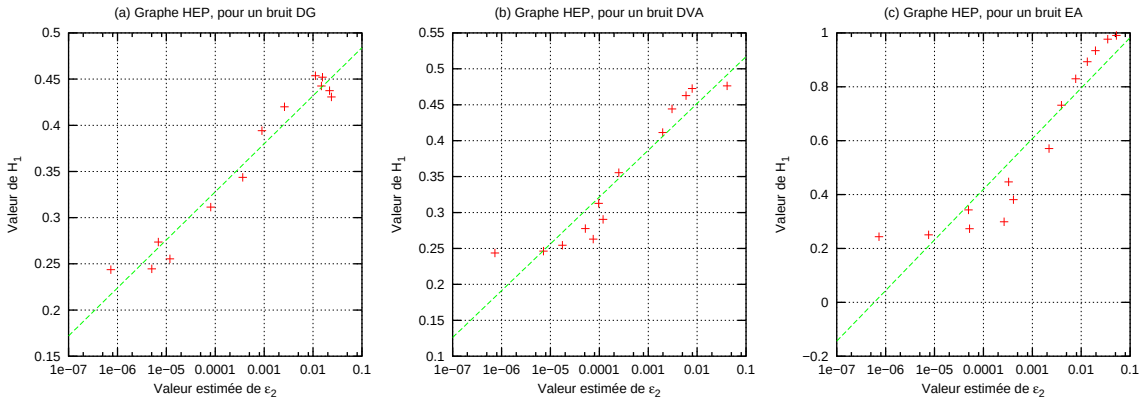
– la topologie des états du système

Dans le cas du graphe (a), les données d'entrée sont parfaites, ce qui élimine la première cause d'erreurs. La variation de la performance d'un essai à l'autre n'existe que parce que l'état initial du système change, entraînant des trajectoires différentes d'un essai à l'autre. Bien évidemment, si on choisit toujours le même état initial pour le système, et qu'on n'introduit pas de bruit de mesure, les trajectoires d'un essai à l'autre sont identiques au bout d'un certain nombre d'essais (lorsque la probabilité de ne pas appliquer une politique gloutonne devient trop faible).

Abordons, à présent, l'influence d'une dégradation des données sur la durée de viabilité du système. Les graphes (d), (e) et (f) montrent l'évolution de la durée moyenne et de la durée maximum de viabilité en fonction des trois types de bruit de mesure : (d) pour un bruit de mesure gaussien, (e) pour un taux de valeurs aberrantes de θ et (f) pour un taux de choix aléatoire de l'état courant. Les écarts-types sont représentés par les segments verticaux. **Nous constatons que le lien fonctionnel entre la durée de viabilité moyenne, sur les 2000 essais d'un apprentissage, et l'amplitude du bruit de mesure peut être modélisée assez précisément par une fonction du type** $exp(b).\tau^a$, τ représentant l'amplitude du bruit de mesure, « a » la pente de la droite figurant sur les graphes (d), (e) et (f) en log/log, et « b » la valeur de cette droite à l'origine. Les valeurs du paramètre « a » sont peu différentes suivant le type de bruit de mesure (voir les commentaires en bas de la figure 1.11). Donc, si on accepte l'hypothèse (H) et qu'on utilise une modélisation à deux sources d'erreurs, on en déduit que la valeur de ϵ_2 croît en fonction de l'amplitude du bruit de mesure, suivant une fonction équivalente à $exp(-b).\tau^{-a}$. Nous pouvons donc valider partiellement notre interprétation des causes d'erreurs associées à la source E_2. Nous disons « partiellement » car les graphes (g),(h) et (i) montrent que la valeur de ϵ_2 n'est pas constante dans l'absolu (pour un contexte d'apprentissage donné), mais **constante par morceaux**. L'algorithme de choix de la commande semble être à nouveau la cause de cette instabilité. Pour comprendre ce phénomène, il suffit de regarder les conséquences pratiques de l'application d'un bruit de mesure : la qualité du choix d'une commande pour un état donné est modifiée en fonction de celle de l'état d'arrivée (après exécution de la commande). Or, le bruit peut induire le système en erreur sur **la nature exacte de l'état d'arrivé**, dont la qualité peut être très différente de celle de l'état d'arrivée réel (que le système aurait découvert s'il n'y avait pas de bruit de mesure). Une mauvaise qualité peut donc être rétro-propagée et l'effet néfaste est amplifié par la trace d'éligibilité (qui va modifier d'une manière erronée l'ensemble des états dans lequel le système s'est trouvé depuis l'initialisation de l'essai).

En résumé, nous avons montré les points suivants :

– l'évolution de la durée de viabilité du système au cours des essais d'apprentissage peut être modélisée par deux flux d'erreurs E_1 et E_2, suivant le modèle établi dans la sous-section 1.3.11.
– le flux d'erreurs E_1 est caractérisé par un paramètre ϵ_1 élevé, de l'ordre de 5.10^{-3}, possédant une faible variabilité suivant les apprentissages, les types de bruit de mesure et l'amplitude de ces bruits. Il est causé principalement par un mauvais paramétrage du dilemme exploration/exploitation gérant le choix d'une commande à tout moment, qui est un facteur du contexte de l'apprentissage (le paramètre lié à la résolution du dilemme est régi par une loi d'évolution indépendante de l'état de l'apprentissage). La probabilité p qu'une erreur soit provoquée par E_1 est variable selon les apprentissages et n'est pas constante dans un même apprentissage (elle est constante par morceaux).
– le flux d'erreurs E_2 est caractérisé par un paramètre ϵ_2, dont la valeur est influencée par la



Légende :
« HEP » Graphe de la relation entre ϵ_2 (abscisses) et H_1 (ordonnées), suivant différentes amplitudes d'un même type de bruit de mesure. L'échelle logarithmique est appliquée pour les abscisses.
« DG » Données issues de θ bruitées avec un bruit gaussien, d'amplitude σ.
« DVA » Données issues de θ possédant un taux de valeurs aberrantes τ_{VA}.
« EA » L'état du système peut être choisi aléatoirement, avec une fréquence d'apparition τ_{EA}.
Commentaires : Les points des graphes (a), (b) et (c) sont obtenus en fixant une amplitude de bruit de mesure et en récupérant les valeurs de H_1 et de ϵ_2 associées. L'évolution de H_1 est traitée dans la sous-section 1.4.4, alors que l'étude de ϵ_2 est donnée dans la sous-section 1.4.5. Les droites sont obtenues en utilisant le critère des moindres carrés, par l'algorithme de Levenberg-Marquardt. Elles modélisent un lien fonctionnel, pour lequel H_1 peut s'exprimer en fonction de ϵ_2 : $H_1 = \text{a.log}(\epsilon_2)+\text{b}$. Dans ce cadre, on trouve, pour les types de bruit DG, DVA et EA, les valeurs suivantes de a et b : a=0.023, b=0.54 ; a=0.028, b=0.58 ; a=0.082, b=1.17.

Fig. 1.12 – Graphes associés à la sous-section 1.4.6

topologie des états du système et par l'amplitude et le type du bruit de mesure. La valeur de ϵ_2 est constante par morceaux sur un apprentissage.

Ces points montrent que l'hypothèse (H) est valide. On en déduit que la politique de commande résultant d'un apprentissage n'est pas fiable en général : lorsqu'il existe un bruit de mesure, même faible, les valeurs maximales de la durée de viabilité sont inférieures à 100 millions d'itérations dans **tous les cas**, ce qui relativise le résultat de Pendrith sur la réussite de pratiquement tous les essais qu'il a effectués (mais celui-ci utilise 10.000 itérations). D'une certaine manière, on peut caractériser la nature des durées de viabilité (existence de deux bandes). Par contre, pour un essai donné, on ne peut pas prédire avec certitude la durée de viabilité qui sera obtenue (on n'obtient qu'un résultat statistique). Nous avons montré, pour notre modélisation, que **ce manque de prédictibilité** provient de l'influence de l'algorithme qui gère le dilemme exploration/exploitation, corrélée avec le choix aléatoire de l'état initial du système.

1.4.6 Relation entre H_1 et ϵ_2

Dans la sous-section 1.4.4, nous avons montré que H_1 pouvait mesurer la qualité du contexte d'apprentissage. Puis, dans la sous-section 1.4.5, nous avons établi que le contexte d'apprentissage pouvait être modélisé suivant le modèle établi dans la sous-section 1.3.11, introduisant ainsi trois paramètres : p, ϵ_1 et ϵ_2. Nous avons relié la valeur de ϵ_2 avec l'amplitude du bruit de mesure appliqué aux entrées du système. À présent, et pour clore notre étude sur l'influence du contexte sur le résultat de l'apprentissage, il nous faut montrer la relation qui existe entre la mesure H_1 et ϵ_2. Les graphes (a), (b) et (c) de la figure 1.12 dévoilent l'existence d'un tel lien, suivant les trois

catégories de bruit de mesure. Il apparaît clairement que **ce lien est « grossièrement » fonctionnel et bijectif**. Cela signifie que, si on connaît la catégorie de bruit de mesure appliquée au système, on peut théoriquement utiliser H_1, **déterminé avant l'expérience d'apprentissage**, pour donner une prédiction approximative sur la valeur de ϵ_2, donc sur le comportement global des durées de viabilité de la seconde source d'erreurs, entretenue entre autres par l'amplitude du bruit de mesure. Nous pouvons même préciser un modèle correct de cette relation : les droites tracées sur les graphes (a), (b) et (c), dont les abscisses sont logarithmiques, montrent une relation du type : $H_1 = \text{a.log}(\epsilon_2)+\text{b}$. Nous remarquons que les valeurs de a et de b sont très proches pour les graphes (a) et (b) (voir les commentaires au bas de la figure 1.12). Cela signifie qu'**un même lien fonctionnel existe entre H_1 et ϵ_2, pour des bruits gaussien sur θ et un taux de valeurs aberrantes de θ.**

L'utilisation de H_1 permet de calculer, dans une certaine mesure, la valeur de ϵ_2, donc de **prédire, avant l'apprentissage, la répartition des durées de viabilité dues à la source d'erreurs E_2, obtenues après apprentissage.**

1.4.7 Conclusion

Le banc d'essai du pendule inversé nous a permis de mettre en évidence certains mécanismes d'interactions entre l'algorithme d'AR et son contexte. Dans le cas du pendule inversé, **nous avons pu valider l'hypothèses (H)**, proposée dans la sous-section 1.3.9. Nous avons montré que le modèle proposé dans la sous-section 1.3.11 est relativement adapté pour expliquer la répartition des durées de viabilité au cours d'un apprentissage. L'existence d'une telle modélisation implique que **l'apprentissage ne garantit pas d'obtenir une politique de commande fiable** et que, au début d'un essai d'apprentissage, **on ne peut pas prédire** la nature de son résultat. Néanmoins, on peut caractériser statistiquement les performances de l'apprentissage, en regardant la répartition des durées de viabilité. Nous avons mis en évidence l'existence de deux zones dans lesquelles les durées de viabilité sont cantonnées : chaque zone est associée avec une des deux sources d'erreurs prévues par notre modèle. Nous avons caractérisé la nature probable des deux sources d'erreurs E_1 et E_2 : E_1, regroupant les essais dont la durée de viabilité est courte (autour de 150 itérations), est probablement due au paramétrage qui règle le dilemme exploration/exploitation, alors que la deuxième source d'erreurs dépend à la fois de la topologie des états du système et de la qualité des données en entrée de celui-ci.

Nous avons complété les observations de Pendrith [Pendrith, 1994] et modifié certaines conclusions qu'il avait apportées. En particulier, lorsqu'on augmente beaucoup le nombre d'itérations limite définissant qu'un apprentissage est réussi, on s'aperçoit que peu d'apprentissages le sont et que l'ajout d'un bruit de mesure, même faible, empêche systématiquement la réussite de l'apprentissage.

Enfin, nous avons montré l'utilité des mesures H_1 et H_2, qui permettent effectivement de caractériser la qualité du contexte d'apprentissage et l'influence (ou le peu d'influence) des variables d'entrée. Nous avons prouvé l'existence d'un lien grossièrement fonctionnel et bijectif qui unit H_1 et la valeur de ϵ_2 relatif à la deuxième source d'erreurs : $H_1 = \text{a.log}(\epsilon_2)+\text{b}$. Ce lien permet, en théorie, de prévoir l'évolution statistique des durées de viabilité, en connaissant la nature du bruit de mesure et la valeur de H_1, qu'on peut déterminer avant le début de l'apprentissage : plus H_1 est faible et plus ϵ_2 est faible, donc plus la durée de viabilité moyenne est importante. Nous avons aussi montré, grâce à l'évolution de H_2 et de H_1, que la variable θ est peu discriminante

et qu'on peut découper cet axe moins finement que ce qui est choisi d'habitude, ce qui n'est pas un résultat évident *a priori*.

Références

Albus, J. (1971). A theory of cerebellar function. *Mathematical Biosciences*, 10 :25–61.

Albus, J. (1981). *Brain, Behavior, and Robotics*. Byte Books.

Baird, L. C. (1995). Residual algorithms : Reinforcement learning with function approximation. *Proceedings of the Twelfth International Conference on Machine Learning*, pages 30–37.

Barto, A., Sutton, R., and Anderson, C. (1983). Neurolike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC13 :834–846.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press, Princeton, NJ.

Bersekas, D. and Tsitsiklis, J. (1996). *Neuro-dynamic Programming*. Athena Press.

Bersini, H. and Gorrini, V. (1996). Three connectionist implementations of dynamic programming for optimal control : A preliminary comparative analysis. In *Workshop on Neural Networks for Identification and Control in Robotics*.

Bertsekas, D. (1987). *Dynamic Programming*. Prentice Hall.

Bouchon, B. (1988). Entropic models : a general framework for measures of uncertainty and information. *Logic in Knowledge-Based Systems, Decision and Control*, pages 93–105.

Bouchon-Meunier, B. (1989). Incertitude, information, imprÃ©cision : une rÃ©flexion sur l'Ã©volution de la thÃ©orie de l'information. *Revue Internationale de SystÃ©mique*, 3(4) :375–385.

Davesne, F. and Barret, C. (1999). Reactive navigation of a mobile robot using a hierarchical set of learning agents. In *IROS'99*, Kyongyu, Corée.

Dayan, P. (1992). The convergence of td(λ) for general λ. *Machine Learning*, 8 :341–362.

Dayan, P. and Sejnowski, T. (1994). Td(λ) converges with probability 1. *Machine Learning*, 14 :295–301.

Hartley, R. (1928). Transmission of information. *Bell System technical Journal*, 7 :535–563.

Jouffe, L. (1997). *Apprentissage de systÃ¨mes d'infÃ©rence floue par des mÃ©thodes de renforcement : application Ã la rÃ©gulation d'ambiance dans un bÃtiment d'Ã©levage porcin*. Thèse de doctorat, UniversitÃ© de Rennes 1.

Kaelbling, L., Littman, M., and Moore, A. (1996). Reinforcement learning : A survey. *Journal of Artificial Intelligence Research*, 4 :237–285.

Lin, C. and Kim, H. (1991). Cmac-based adaptive critic self-learning control. *IEEE Transactions on Neural Networks*, 2 :530–533.

Lin, L.-J. (1992). Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8 :293–321.

Lovejoy, W. (1991). A survey of algorithmic methods for partially observable markov decision processes. *Annals of Operations Research*, 28 :47–65.

McGovern, A., Precup, D., Ravindran, B., Singh, S., and Sutton, R. (1998). Hierarchical optimal control of mdps. *Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems*, pages 186–191.

Munos, R. (1997). *Apprentissage par Renforcement, Étude du cas Continu.* Thèse de doctorat, EHESS, CEMAGREF.

Nowé, A. (1995). Fuzzy reinforcement learning : an overview. *Advances in fuzzy theory and technology.*

Pendrith, M. (1994). On reinforcement learning of control actions in noisy and non-markovian domains. Technical report, UNSW Computer Science and Engineering.

Pendrith, M. (1999). Reinforcement learning in situated agents : Some theoretical problems and practical solutions. In *8th European Workshop on Learning Robots.*

Pendrith, M. and McGarity, M. (1998). An analysis of direct reinforcement learning in non-markovian domains. *The Fifteenth International Conference on Machine Learning.*

Peng, J. and Williams, R. (1996). Incremental multi-step q-learning. *Machine Learning*, 22 :283–290.

Quinlan, J. (1984). Learning efficient classification procedures and their application to chess endgames. *Machine Learning. An Artificial Approach*, pages 463–482.

Rummery, G. (1995). *Problem Solving with Reinforcement Learning.* PhD thesis, Cambridge University.

Samuel, A. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3 :211–229.

Schwartz, A. (1993). A reinfocement learning method for maximising undiscounted rewards. *Proceeding of Int. Conf. on Machine Learning.*

Shannon, C. (1948). A mathematical theory of communication. *Bell System technical Journal*, 27 :379–423,623–656.

Sutton, R. (1984). *Temporal Credit Assignment in Reinforcement Learning.* PhD thesis, University of Massachusetts, Amherst, MA.

Sutton, R. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, 3 :9–44.

Sutton, R. (1996). Generalization in reinforcement learning : Successful examples using sparse coarse coding. *Advances in Neural Information Processing Systems : Proceedings of the 1995 Conference*, pages 1038–1044.

Sutton, R. and Barto, A. (1998). *Reinforcement Learning : An introduction.* MIT Presss, Cambridge, MA.

Tesauro, G. (1994). Td-gammon, a self-teaching backgammon program, achieves masterlevel play. *Neural Computation*, 6 :215–219.

Tsitsiklis and Roy, V. (1996). Feature-based methods for large scale dynamic programming. *Machine Learning*, 22 :59–94.

Tsitsiklis, J. (1994). Asynchronous stochastic approximation and q-learning. *Machine Learning*, 16 :185–202.

Watkins, C. (1989). *Learning from Delayed Rewards.* PhD thesis, King's College, Cambridge, UK.

Wiering, M. (1999). *Explorations in Efficient Reinforcement Learning.* PhD thesis, Universiteit van Amsterdam.

2

Algorithme d'AO défini dans un contexte idéal

2.1 Introduction

2.1.1 Idée directrice du chapitre

Dans notre avant-propos, nous avons proposé une modélisation d'un système apprenant capable d'apprentissage d'actions réflexes. Cette modélisation rassemble **une démarche** (§3.2) et **un modèle** (figure 2, page xxiii). Ce dernier est séparé en deux sous-systèmes : un sous-système d'apprentissage perceptif et un sous-système d'apprentissage d'objectif. Ce chapitre étudie ce dernier, en utilisant notre démarche.

Cette dernière impose que l'évolution du système soit due à la réaction de celui-ci à l'action de l'environnement, dans le but de respecter des contraintes internes.

Nous avons montré dans le chapitre précédent que le contexte de l'algorithme d'apprentissage influe sur la qualité de l'apprentissage, rendant la nature du résultat obtenu peu prédictible et la fiabilité de celui-ci incertaine. Notre objectif est d'établir un algorithme dont le seul contexte est la topologie des états du système, construite à partir de l'apprentissage perceptif. En particulier, **nous souhaitons éviter l'introduction de paramètres internes, non accessibles à l'algorithme en lui-même, dont la valeur pourrait influencer la nature du résultat de l'apprentissage**.

Nous modélisons le sous-système d'apprentissage d'objectif par un graphe d'états, dont chacun d'entre-eux possède un marquage (ou une *Q-value*, si on utilise un vocabulaire inspiré de l'apprentissage par renforcement). Les contraintes s'expriment par une relation liant les marquages de nœuds voisins dans le graphe (reliés par un arc). Elles sont similaires à celles employées dans un algorithme classique d'IA : l'algorithme *Minimax* [Rich, 1983].

L'action de l'environnement sur le sous-système s'effectue par la création de transitions entre les états du graphe. Celle-ci correspond à la génération d'un modèle interne de l'environnement et constitue un **apprentissage latent**.

La réaction du sous-système s'effectue **lorsqu'une nouvelle transition est détectée**, par

modification (le cas échéant) des valeurs des marquages (phénomène de propagation).

2.1.2 Principaux résultats

À partir de ce modèle, nous en déduisons l'ensemble des évolutions possibles du sous-système, les preuves concernant le rétablissement des contraintes à chaque pas de temps, puis la convergence des marquages.

Nous en déduisons une **condition suffisante** pour que l'évolution des marquages puisse être interprétée comme un apprentissage. Celle-ci est caractérisée par un **contexte d'apprentissage particulier**, que nous appelons *contexte d'apprentissage idéal*, qui respecte la propriété (P_ϵ). Ce dernier est caractérisé par des mesures H_1 et H_2 très proches de 0 (voir le chapitre 1).

À partir des contraintes, nous spécifions un algorithme, appelé **CbL** pour *Constraint based Learning*, dont nous montrons les caractères de fiabilité et de prédictibilité.

Deux applications illustrent nos résultats théoriques : l'exemple jouet du labyrinthe et un problème de navigation d'un robot mobile simulé. Elles montrent les capacités d'incrémentalité de CbL et la rapidité de l'apprentissage, en comparaison avec une méthode d'AR classique ($Q(\lambda)$ [Peng et Williams, 1996]). Cela est une conséquence directe de l'utilisation de l'apprentissage latent [Stolzmann, 1998].

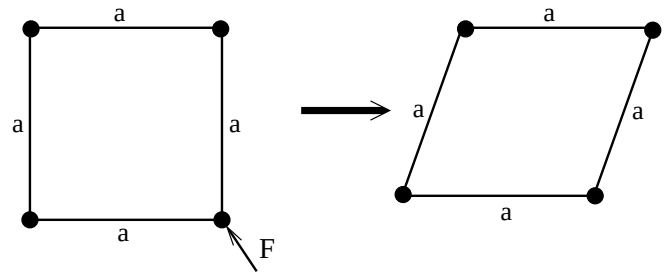
2.1.3 Plan du chapitre

Ce chapitre est consacré à la construction d'un algorithme, appelé CbL, qui respecte la méthodologie que nous nous sommes imposée. La section 2.2 décrit le système et les contraintes appliquées à celui-ci. L'algorithme CbL est donné dans la section suivante (voir l'algorithme 2.1 à la page 45). L'étude théorique est détaillée dans la section 2.4. Nous prouvons que l'ensemble de nos exigences est respecté, lorsque la propriété (P_ϵ) est satisfaite (atteinte d'objectif) ou lorsque le problème de choix de commande est markovien (respect d'une zone de viabilité). Les propriétés d'incrémentalité de CbL sont discutées dans la section 2.5. Les deux sections suivantes étudient des applications de l'algorithme CbL. Le problème du labyrinthe, dont on sait qu'il respecte la propriété (P_ϵ), sert d'exemple applicatif d'atteinte d'objectif (section 2.6). Nous étudions également un problème de navigation d'un robot miniature Khepera simulé (section 2.7) : l'apprentissage d'un comportement de suivi de mur est un exemple simple de respect d'une zone de viabilité. Dans ces exemples, nous comparons les performances de l'algorithme CbL avec celles du Q-Learning. Le lecteur pourra consulter les travaux faisant référence à ce chapitre : [Davesne et Barret, 1999b], [Davesne et Barret, 1999a], pour une description de l'algorithme d'AO, ainsi que pour son application dans une tâche de navigation d'un robot mobile Khepera simulé.

2.2 Modélisation du sous-système d'apprentissage d'objectif

2.2.1 Méthodologie

En pratique, notre méthodologie est directement inspirée de l'étude d'un système physique soumis à un ensemble de forces (qui sont, dans notre cas, appelées « contraintes »). Notre analyse



L'application d'une force F sur l'objet composé de quatre tiges rigides de longueur a, en liaison rotule, provoque une modification de la forme de cet objet. Mais celui-ci sera toujours un parallélogramme (invariant structurel).

Fig. 2.1 – Exemple d'invariant structurel en mécanique.

du problème d'apprentissage utilise trois étapes successives :

1. spécification du système[1] et des contraintes d'équilibre qui s'appliquent à ce système, à tout instant
2. spécification de l'interaction entre le système et son environnement
3. preuve que l'interaction peut être interprétée comme un apprentissage

Un système auquel des contraintes d'équilibre s'appliquent peut être imagé par un solide déformable (voir la figure 2.1) : il possède des degrés de liberté, mais les possibilités de déformation du système sont limitées par les contraintes. Cela se traduit, dans le cas de la figure 2.1, par l'existence d'un invariant structurel de l'objet global : il s'agit toujours d'un parallélogramme, quelle que soit la nature de la force F.

Pour appliquer notre méthodologie, il faut préciser les points suivants :

- la nature du système étudié
- les contraintes appliquées à ce système
- l'action de l'environnement sur le système
- la réaction du système suite à l'action de l'environnement

2.2.2 Spécification du sous-système

Le sous-système est un **graphe d'états**, défini de la manière suivante.

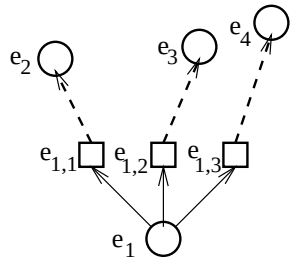
Il existe un nombre fini d'états e_1,e_2,...,e_n, dont on suppose qu'ils sont spécifiés *a priori*[2].

À tout instant, le sous-système reçoit deux informations :

1. l'état courant e_i
2. un signal de renforcement r, qui prend ses valeurs dans l'ensemble $\{0,1,-1\}$. Lorsque r vaut 1, cela signifie que l'objectif est atteint ; lorsque r vaut -1, le système est sorti de sa zone de viabilité ; dans les autres cas, r vaut 0.

À partir de ces deux informations, le sous-système décide d'exécuter une action parmi les q possibles, contenues dans un ensemble A spécifié *a priori*. $A = \{a_1,a_2,...,a_q\}$

1. Ce système est celui qui doit montrer des propriétés d'apprentissage. 2. Le rôle de l'apprentissage perceptif est de construire cet ensemble d'états.



Les états perceptifs e_i sont représentés par des cercles, alors que les états transitoires $e_{i,k}$ sont représentés par des carrés. Dans cet exemple, le système possède trois actions. À l'instant t, le système est dans l'état e_1, qui est lié à trois états transitoires (dépendant du choix entre trois actions possibles). Les transitions (en pointillé) sont le résultat des expériences passées du système, liant l'exécution de chacune de ces actions à un certain nombre d'états perceptifs.
La propriété (P_ϵ) se traduit sur ce graphe par l'existence, en pratique, d'une unique transition entre un état transitoire et un état perceptif.

FIG. 2.2 – Les catégories d'états du système.

Le graphe possède deux autres catégories d'états :

- deux états « terminaux » e_S et e_E. Lorsque r vaut 0, le système se trouve dans un des états e_1,e_2,...,e_n, alors que si r vaut 1, le système se trouve dans l'état terminal e_S et si r vaut -1, le système se trouve dans l'état terminal E_E
- l'exécution d'une action a_k à partir de l'état e_i, à l'instant t, amenant à l'état e_j à l'instant t+1, se traduit par le passage du système dans un état transitoire, noté $e_{i,k}$, entre les instants t et t+1. Donc, entre les instants t et t+1, deux transitions sont franchies : l'une de e_i vers $e_{i,k}$ et l'autre de $e_{i,k}$ vers e_j (figure 2.2)

Chaque état de type e_i ou de type $e_{i,k}$ est associé à **un marquage**, nommé M_i pour e_i et $M_{i,k}$ pour $e_{i,k}$. Ces marquages prennent leur valeur dans un ensemble SM fini. Nous considérons l'ensemble SM suivant :

$$SM = SM(\alpha,d) = \{-1,0,1,\alpha,\alpha^2,...,\alpha^d\}$$

Avec $d \geq 1$ et $\alpha \in]0,1]$. Nous considérerons que lorsque $\alpha = 1$, $SM = SM(1) = \{-1,0,1\}$.

Les états E_S et E_E possèdent un marquage fixe (respectivement égal aux symboles 1 et -1). Le marquage associé aux autres états (les e_i et les $e_{i,k}$) va être déterminé par des modifications du sous-système (pour respecter ses contraintes) et sera égal à l'un des symboles de SM.

2.2.3 Contraintes appliquées au sous-système

Il nous faut à présent spécifier les contraintes appliquées au sous-système. Elles sont formées à partir d'une relation d'invariance inspirée de l'algorithme *Minimax* [Rich, 1983], liant des états voisins du graphe.

Voici la spécification des contraintes, valables pour chaque état, à chaque pas de temps. Celles-ci précisent la valeur du marquage M_i de chaque état perceptif e_i ainsi que le marquage $M_{i,k}$ de chaque état transitoire $e_{i,k}$, en fonction du marquage des états connectés à ces derniers :

$$\begin{cases} \forall i \in \{1,...,n\},\ M_i = \max_{j\in\{1,...,q\}}\{M_{i,j}\} \\ \forall (i,k) \in \{1,...,n\}\{1,...,q\},\ M_{i,k} = \alpha.\min_{j\in V(M_{i,k})}\{M_j\} \end{cases} \quad (2.1)$$

$V(M_{i,k})$ est l'ensemble des indices des états e_j vers lesquels une transition issue de l'état transitoire $M_{i,k}$ aboutit. On supposera que si $V(M_{i,k})$ est vide, alors $M_{i,k}$ vaut 0. Cette supposition est importante pour que l'ensemble des marquages initiaux soient cohérents (voir la sous-section 2.3). Dans le cas où $\alpha < 1$, $\alpha(M_j)$ vaut 1 lorsque M_j est égal à -1 ou 0, est égal à $\alpha \in]0,1]$ si $M_j \in \{1,\alpha,...,\alpha^{d-1}\}$ et est égal à 0 si $M_j = \alpha^d$. Dans le cas particulier où $\alpha = 1$, on fixera $\alpha(M_j)$ à 1, quel que soit M_j. La contrainte appliquée au graphe d'états lie les valeurs des marquages associés à des états voisins (reliés par une transition).

2.2.4 Action de l'environnement et réaction du sous-système

L'action de l'environnement se traduit par **l'ajout de transitions dans le graphe d'états** entre les états $e_{i,k}$ et les états e_j.

La réaction du sous-système consiste à modifier les marquages associés à chacun des états e_i et $e_{i,k}$ de manière à ce que la relation d'invariance 2.1 reste valable à chaque instant.

L'ensemble action/réaction donne naissance à l'algorithme CbL (§2.3).

2.3 Algorithme d'apprentissage Constraint based Learning (CbL)

2.3.1 Introduction

Le déroulement d'un apprentissage comporte les mêmes phases que pour l'AR (voir le chapitre 1). Nous ne redéfinirons pas les termes « itération », « essai » et « apprentissage », spécifiés dans le chapitre précédent. Il nous faudra cependant préciser les points suivants :

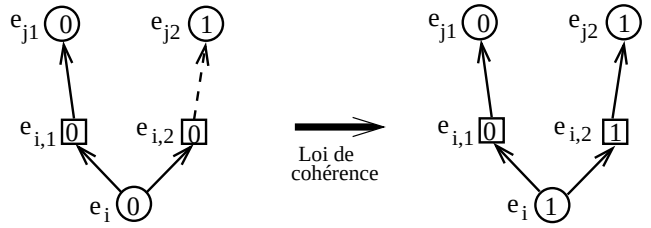
- apprentissage : comment modifier le marquage associé à chaque état du système
- initialisation : comment initialiser les marquages ?
- choix d'une action

2.3.2 Exemple de propagation des marquages du graphe d'état

L'idée pour fixer le marquage des états est de rétro-propager les valeurs des états terminaux par l'intermédiaire des transitions existantes. Ce mécanisme est fondé sur l'existence d'un ensemble de contraintes (équation 2.1).

Si on suppose que la contrainte d'équilibre est respectée à un instant t pour chaque état, seul l'ajout d'une transition dans le graphe d'états (action de l'environnement) peut le rendre incohérent. La figure 2.3 illustre ce fait. Dans ce cas, il faut modifier le marquage associé à l'état incohérent. Le rétablissement de la cohérence sur cet état peut engendrer une incohérence sur au moins un des états en contact avec l'état incohérent. Il faut donc modifier également leur marquage : **il y a ici un mécanisme de propagation**.

Nous montrerons, dans la partie théorique de ce chapitre, que ces modifications par propagation des marquages M_i et $M_{i,k}$ peuvent être interprétées comme un apprentissage.



La figure de gauche montre des états e_1, $e_{i,1}$ et $e_{i,2}$ dont les marquages sont cohérents. L'ajout d'une transition entre $e_{i,2}$ et e_{j2} rompt cette cohérence, puisque le marquage de $e_{i,2}$ vaut 0 et n'est plus égal au minimum des marquages associés aux états voisins (e_{j2}). La figure de gauche montre le graphe après le rétablissement de la cohérence entre les marquages. On note que la valeur de M_i a également été modifiée. Ce changement pourrait, à son tour, provoquer un changement des marquages associés aux états connectés à e_i, qui n'apparaissent pas sur le schéma.

Fig. 2.3 – L'apprentissage est provoqué par l'ajout d'une transition dans le graphe d'états.

2.3.3 Algorithme CbL

L'apprentissage correspond à la mise en cohérence du graphe d'états, grâce au mécanisme de propagation décrit ci-dessus. Les détails de ce mécanisme d'apprentissage sont donnés par l'algorithme 2.1, s'appuyant sur les relations 2.1.

Deux points de l'algorithme d'apprentissage restent à aborder :

- l'initialisation des marquages
- le choix des actions

Pour ce qui concerne l'initialisation, au début de l'apprentissage (avant le premier essai), on considère qu'il n'existe aucune transition entre les sommets $e_{i,k}$ et e_i du graphe. Le marquage associé à chaque état est donc 0. Cette initialisation respecte la contrainte d'équilibre donnée par l'équation 2.1. Les transitions sont ajoutées au fur et à mesure de leur découverte, au cours des essais de l'apprentissage du système. L'ajout d'une transition peut provoquer le non respect de cette contrainte d'équilibre ; si cela survient, on utilise l'algorithme 2.1 pour rétablir la cohérence.

Enfin, si le système est dans l'état e_i, l'action sera choisie selon la valeur du marquage $M_{i,k}$ lui étant associé. Nous utiliserons une politique « gloutonne » : on sélectionnera l'action pour laquelle la valeur du marquage est maximum. Si plusieurs actions possèdent le même marquage de valeur maximum, on en choisira une au hasard parmi celles-ci.

Une vue de haut niveau du processus complet d'apprentissage CbL est donnée par l'algorithme 2.2. Cet algorithme est celui d'un essai d'apprentissage, qui se termine lorsque l'objectif est atteint (r=1), la zone de viabilité est quittée (r=-1) ou qu'un nombre maximum d'itérations est dépassé.

2.3.4 Remarques

Le problème lié au choix des marquages des états e_i et $e_{i,k}$ provient de l'ignorance *a priori* de l'effet des actions sur le passage d'un état à l'autre (on suppose qu'on ne possède pas de modèle de la dynamique du problème).

Le choix des équations de l'algorithme **Minimax** s'interprète de la manière suivante. Nous

Algorithme 2.1 Algorithme de mise en cohérence des marquages M_i et $M_{i,k}$

Si une transition est créée entre l'état transitoire $e_{i,k}$ et l'état perceptif e_j, alors
    Appeler la fonction CohérenceIK($e_{i,k}$,e_j)
FinSi

Étape 1 : Test de cohérence au niveau de $e_{i,k}$
CohérenceIK($e_{i,k}$,e_j)
    Si $M_j \neq \alpha.\inf_{j' \in V(M_{i,k})}\{M_{j'}\}$, alors
        $M_{i,k} := \alpha.M_j$
        Si $0 < M_{i,k} < \alpha^d$, alors
            $M_{i,k} := 0$ (Il n'existe plus de marquage disponible)
        FinSi
        Appeler la fonction CohérenceI(e_i) (propagation au noeud perceptif e_i)
    Sinon,
        C'est fini (pas d'incohérence détectée)
    FinSi

Étape 2 : Test de cohérence au niveau de e_i
CohérenceI(e_i)
    Si $M_i \neq \max_{k' \in \{1,\ldots,q\}}\{M_{i,k'}\}$
        $M_i := \max_{k' \in \{1,\ldots,q\}}\{M_{i,k'}\}$
        (Phase de propagation)
        Pour tous les $e_{j,k}$ possédant une transition vers e_i, faire
            Appeler la fonction CohérenceIK($e_{j,k}$, e_i)
        FinPour
    Sinon,
        C'est fini (cohérence obtenue pour l'état e_i)
    FinSi

Algorithme 2.2 Algorithme CbL de haut niveau

Pour tous les états e_i et $e_{i,k}$, faire
    $M_i := 0$, $M_{i,k} := 0$
FinPour
Récupérer l'état initial e_i
Répéter
    Choisir l'action a_k dont le marquage $M_{i,k}$ est maximum.
    Si deux actions possèdent le même marquage maximum,
        En choisir une au hasard
    FinSi
    Répéter
        Exécuter l'action a_k^* sélectionnée
        Récupérer la valeur du signal de renforcement r
        Récupérer l'état courant e_j
    Jusqu'à $e_j \neq e_i$ ou r vaut 1 ou -1
    Si r=1, alors
        $e_j := e_S$
    Sinon si r=-1 alors
        $e_j := e_E$
    FinSi
    Si la transition entre e_i et e_j n'existe pas,
        Créer cette transition
        Appeler l'algorithme 2.1 (apprentissage)
    FinSi
    $e_i := e_j$
Jusqu'à r=1 ou r=-1 ou nombre d'itérations maximum atteint

avons considéré que, dans ce cadre, le choix de l'action peut être vu comme un problème de prise de décision associé à un jeu à deux joueurs : un joueur serait le système (qui possède, à chaque pas de temps, q coups possibles), alors que l'autre joueur serait la dynamique du système. À chaque instant, l'objectif du système est d'empêcher son adversaire (la dynamique) de le pousser vers l'état terminal, ce qui signifierait la perte de la partie. Dans ce cas, l'ensemble des marquages M_i vaudraient -1.

L'action de l'environnement sur le système permet la création d'un modèle interne de l'environnement, grâce à la création des transitions dans le graphe. Cela est interprétable comme un **apprentissage latent**.

Dans notre cas, c'est l'évolution du modèle interne qui conditionne la valeur des marquages donc la politique de choix d'action. En faisant une comparaison avec l'AR, on peut dire que cet apprentissage latent est comparable à la phase d'exploration de l'AR et que l'exploitation débute dès qu'un objectif est détecté. Nous montrerons cela dans la partie applicative de ce chapitre.

2.4 Résultats théoriques concernant l'algorithme CbL(1)

2.4.1 Problèmes théoriques

Nous abordons à présent l'aspect théorique de l'algorithme CbL proposé dans la section précédente. À ce sujet, plusieurs questions se posent :

1. la phase de propagation se termine-t-elle toujours et rend-elle le graphe cohérent ?
2. l'algorithme d'apprentissage converge-t-il ?
3. la politique de choix d'action a-t-elle un caractère d'optimalité ?
4. quelle est la fiabilité de la politique de choix d'action après apprentissage ?
5. quelle est la prédictibilité associée à l'algorithme d'apprentissage ?

L'objectif de cette section est de répondre à ces questions.

Dans ce document, nous nous limiterons au cas où la valeur de α est égale à 1, ce qui implique que l'ensemble des marquages possibles est $S = \{0,1,-1\}$. Nous nommerons **CbL(1)** l'algorithme CbL pour cette valeur de α et $CbL(\alpha)$ dans les autres cas. Il faut souligner que les résultats donnés ci-dessous peuvent être facilement étendus au cas $\alpha \in]0,1[$.

1. Nous montrons que la phase de propagation se termine toujours (il n'y a pas de bouclage au cours de la modification des marquages).
2. D'autre part, la convergence de l'algorithme est établie.
3. En utilisant la propriété (P_ϵ), nous montrons que, sous certaines conditions, s'il existe une politique de commande fiable, permettant d'atteindre l'objectif à coup sûr, alors elle est découverte : dans ce sens, la solution obtenue après apprentissage est « optimale », même si ce terme a une portée plus restreinte que pour les méthodes d'AR classique. En effet, toutes les politiques de commande permettant d'atteindre l'objectif sont équivalentes : l'algorithme ne permet pas de les discriminer en dégageant la meilleure. Nous observerons ce point dans l'exemple du labyrinthe fourni dans la prochaine section.

2.4.2 Convergence de la phase de propagation donnée par l'algorithme 2.1

Nous allons montrer que **la phase de propagation se termine toujours** : il n'y a pas de bouclage au cours de la modification des marquages M_i et $M_{i,k}$.

Soit S la somme des marquages de l'ensemble des états e_i et des états transitoires $e_{i,k}$. Considérons à présent la valeur de S à l'instant t, notée $S(t)$, et supposons qu'une nouvelle transition est découverte à cet instant. Admettons que cette transition relie l'état transitoire $e_{i,k}$ à l'état perceptif e_j. La valeur de $M_{i,k}$ dépend donc d'un nouveau marquage M_j.
Deux cas se présentent :

1. l'ajout de M_j ne modifie pas le minimum des marquages des états connectés à $e_{i,k}$
2. la valeur de M_j change le minimum des marquages des états connectés à $e_{i,k}$

Dans le premier cas, l'ajout de cette transition ne modifie pas la cohérence au niveau de $e_{i,k}$ et la phase de propagation n'est pas enclenchée.
Le second cas comporte deux sous-cas : soit la valeur minimum est augmentée par l'ajout du marquage M_j, soit elle est diminuée. Nous allons montrer que, dans les deux cas, la suite des modifications apportées aux marquages par la phase de propagation s'exprime par une évolution monotone de S.

Dans chacun des deux cas, $M_{i,k}$ doit être modifiée pour respecter la cohérence au niveau de l'état $e_{i,k}$. La valeur de $M_{i,k}$ est mise à M_j. Supposons que cette modification rende la nouvelle valeur de $M_{i,k}$ inférieure à la précédente : la nouvelle valeur S(t+1) est alors également inférieure à S(t) après cette modification.

Examinons à présent la cohérence au niveau de l'état e_i. Deux cas de figure existent :

1. la modification de $M_{i,k}$ n'affecte pas le maximum des marquages des états $e_{i,k}$ connectés à l'état e_i
2. la valeur de $M_{i,k}$ change ce maximum

Dans le premier cas, la phase de propagation se termine, avec $S(t+1) < S(t)$.
Dans le second cas, la valeur de M_i est incohérente. Or, le rétablissement de la cohérence, sachant que la nouvelle valeur de $M_{i,k}$ est inférieure à la précédente, se traduit par une diminution de la valeur de M_i, donc par une diminution de S : S(t+2) <S(t+1). Nous reprenons alors l'étape de modification au niveau de chaque état transitoire connecté à e_i. Et chacune de ces étapes conduit soit à une diminution de S, soit à un S inchangé (cas d'arrêt d'une branche de la phase de propagation). Par conséquent, la suite (S(t)) est décroissante. Comme la valeur de S est minorée par -n.(q+1) [1], on en déduit que la suite (S(t)) converge, ce qui signifie que la phase de propagation s'arrête.

Le même raisonnement peut être tenu lorsque la nouvelle valeur de $M_{i,k}$ est supérieure à la précédente. Dans ce cas, la suite (S(t)) est croissante. Or, elle est majorée par n.(q+1), donc elle converge.

Nous venons de montrer que la phase de propagation se termine toujours. Chaque état pour lequel la phase de propagation s'arrête est cohérent, par construction. Par conséquent, le fait de montrer que la propagation se termine est équivalent à dire que le graphe est cohérent après la phase de propagation.

1. n.(q+1) correspond au nombre total d'états, à l'exclusion des états terminaux

2.4.3 Problème d'unicité des valeurs des marquages obtenues grâce à l'algorithme de propagation

Nous allons montrer les points suivants, regroupés sous la forme d'une proposition :

Proposition 1 *dans le cas général, l'historique de l'ajout des transitions influe sur la valeur des marquages obtenus au cours de l'apprentissage :*

- *dans le cas particulier où le système respecte la propriété (P_ϵ), il existe un lien fonctionnel entre le graphe d'états (états et transitions) et la valeur des marquages associés : cela montre l'unicité de l'ensemble des valeurs de marquage obtenues grâce à l'algorithme d'apprentissage*
- *dans le cas particulier où il n'existe aucune transition vers l'état E_S (problème de viabilité), il n'est pas nécessaire que la propriété (P_ϵ) soit respectée pour obtenir ce lien fonctionnel*

Nous allons débuter cette sous-section en montrant les limites du contexte pour lequel il existe une unicité de l'ensemble des valeurs de marquage lorsque le graphe d'états est fixé (états et transitions). La résolution du système d'équations 2.1, pour des valeurs de M_i et $M_{i,k}$ dans 0,1,-1, et un ensemble de transitions fixé, aboutit dans le cas général à plusieurs solutions. La figure 2.4 illustre ce fait, en présentant deux graphes d'états cohérents, possédant les mêmes transitions mais pas les mêmes marquages (graphes (a) et (b)). Pourtant, on sent intuitivement que le graphe (b) ne peut pas être obtenu en utilisant l'algorithme d'apprentissage CbL. En effet, la valeur d'initialisation des marquages étant nulle, on ne voit pas comment il pourrait exister un passage à la valeur 1 sans qu'il existe une transition vers l'état E_S. Cependant, on ne peut pas déduire cela du système d'équations. En effet, le voici :

$$\left\{\begin{array}{c} M_1 = \max\{M_{1,1}, M_{1,2}\} \\ M_2 = \max\{M_{2,1}, M_{2,2}\} \\ M_3 = \max\{M_{3,1}, M_{3,2}\} \\ M_{1,1} = M_2 \\ M_{1,2} = 0 \\ M_{2,1} = 0 \\ M_{2,2} = M_3 \\ M_{3,1} = M_1 \\ M_{3,2} = 0 \end{array}\right\}$$

La résolution de ce système s'effectue de proche en proche. Il vient :

$$\left\{\begin{array}{c} M_1 = M_2 = M_3 = M_{1,1} = M_{2,2} = M_{3,1} \\ M_1 \in \{0,1\} M_{1,2} = M_{2,1} = M_{3,2} = 0 \end{array}\right\}$$

On en déduit qu'il existe deux ensembles de solutions, représentés par le graphe (a) (solutions engendrées en fixant M_1 à 0) et le graphe (b) ($M_1 = 1$).

Le graphe (c) montre par quelle manière on peut passer du schéma (a) au schéma (b) : ajouter une transition vers l'état E_S (en pointillé sur la partie gauche du graphe (c)), donne le schéma de droite du graphe (c) en utilisant l'algorithme de propagation, puis la supprimer redonne le graphe (b), toujours en utilisant l'algorithme de propagation. Or, l'algorithme d'apprentissage ne fonctionne que par ajout de transition. Cet exemple montre que, dans le cas général où on autorise également des suppressions de transitions, la valeur des marquages dépend de l'historique des opérations d'ajout et de suppression.

Les graphes (d) et (e) montrent que, dans le cas général de construction du graphe d'état par apprentissage, la valeur des marquages dépend de l'historique des opérations d'ajout de transitions. Le graphe (d) montre qu'à partir d'un graphe cohérent (partie gauche) dans lequel le marquage des deux états est à 1, l'ajout de la transition de $e_{2,2}$ vers e_1 (en pointillés) ne change pas la cohérence du graphe. D'autre part, le graphe (e) est cohérent, avec le marquage des états valant 0 (figure de gauche) ; l'ajout de la transition de $e_{2,2}$ vers E_S ne change pas la cohérence du graphe. Or, les graphes d'état des deux figures de droite sont identiques, alors que leurs marquages sont différents.

Après avoir fourni ces contre-exemples utiles à une première compréhension des mécanismes de propagation, nous souhaitons montrer que, dans le cadre de l'application de l'algorithme d'apprentissage pour un système vérifiant la propriété (P_ϵ), la valeur des marquages ne dépend pas de l'historique des opérations d'ajout de transitions, conditionné par la stratégie d'exploration des états. Ce résultat est important, car il permet de voir la valeur des marquages obtenue après apprentissage comme une fonction de l'ensemble des transitions découvertes. Par conséquent, la connaissance de ces transitions suffit pour déterminer la valeur des marquages.

Prouvons la validité du deuxième point de la proposition 1, en utilisant les équations de la relation 2.1 liant le marquage des états e_i à celui des états $e_{i,k}$. Pour cela, nous allons établir une formulation modifiée de ce système d'équations.
Commençons par les équations liées aux états transitoires $e_{i,k}$. Celles-ci sont très simplifiées puisque l'hypothèse (P_ϵ) stipule qu'une transition d'un état $e_{i,k}$ ne peut aboutir qu'à un unique état e_j ou à un état terminal. L'expression du marquage $M_{i,k}$ prend quatre formes différentes :

1. $e_{i,k}$ n'est relié à aucun état e_j. Dans ce cas, l'équation associée à $e_{i,k}$ se résume, par hypothèse, à : $M_{i,k} = 0$
2. $e_{i,k}$ est relié à E_E. Dans ce cas, l'équation se résume à : $M_{i,k} = -1$
3. $e_{i,k}$ est relié à E_S. Dans ce cas, il vient : $M_{i,k} = 1$
4. $e_{i,k}$ ne répond à aucun des trois cas précédents. $M_{i,k} = M_j$

Occupons-nous à présent des équations associées aux états e_i. En injectant l'expression des $M_{i,k}$ dans celle de M_i, on trouve quatre cas distincts :

1. un $M_{i,k}$ vaut 1, ce qui entraîne que, quelle que soit l'expression des marquages associés aux autres états $e_{i,k}$, on a l'équation : $M_i = 1$
2. tous les marquages associés aux états transitoires sont fixés à 0 ou -1. Dans ce cas, $M_i =$ c, avec c valant 0 ou -1.
3. les marquages associés aux états transitoires ne comportent que des marquages réductibles à un $M_{j_{i,k}}$ avec éventuellement des marquages valant -1. Dans ce cas, il vient : $M_i = M_{j_i}$, avec $M_{j_i} = \max M_{i,k}$
4. dans les autres cas de figure, il existe des marquages valant 0, ce qui modifie l'expression précédente : $M_i = \max\{0, M_{j_i}\}$

La valeur d'un M_i peut donc être directement connue (cas 1 et 2) ou être en relation avec celle d'un autre marquage (cas 3 et 4). La résolution du système s'effectue soit directement, pour des états qui ne sont pas connectés à d'autres états (cas 1), soit de proche en proche à partir des états connectés aux états terminaux E_S ou E_E. Le problème de l'unicité des valeurs des marquages M_i revient à savoir s'il peut rester des valeurs indéterminées de M_i, répondant aux cas 3 ou 4, à la fin de la résolution de proche en proche. L'exemple que nous avons présenté en début de sous-section indique que cela peut arriver. Dans ce cas, considérons l'ensemble M des marquages ne pouvant être résolus : M = $\{M_{i_1}, M_{i_2}, ..., M_{i_n}\}$. Pour chaque $M_{i_j} \in M$, les cas 3 et 4 imposent

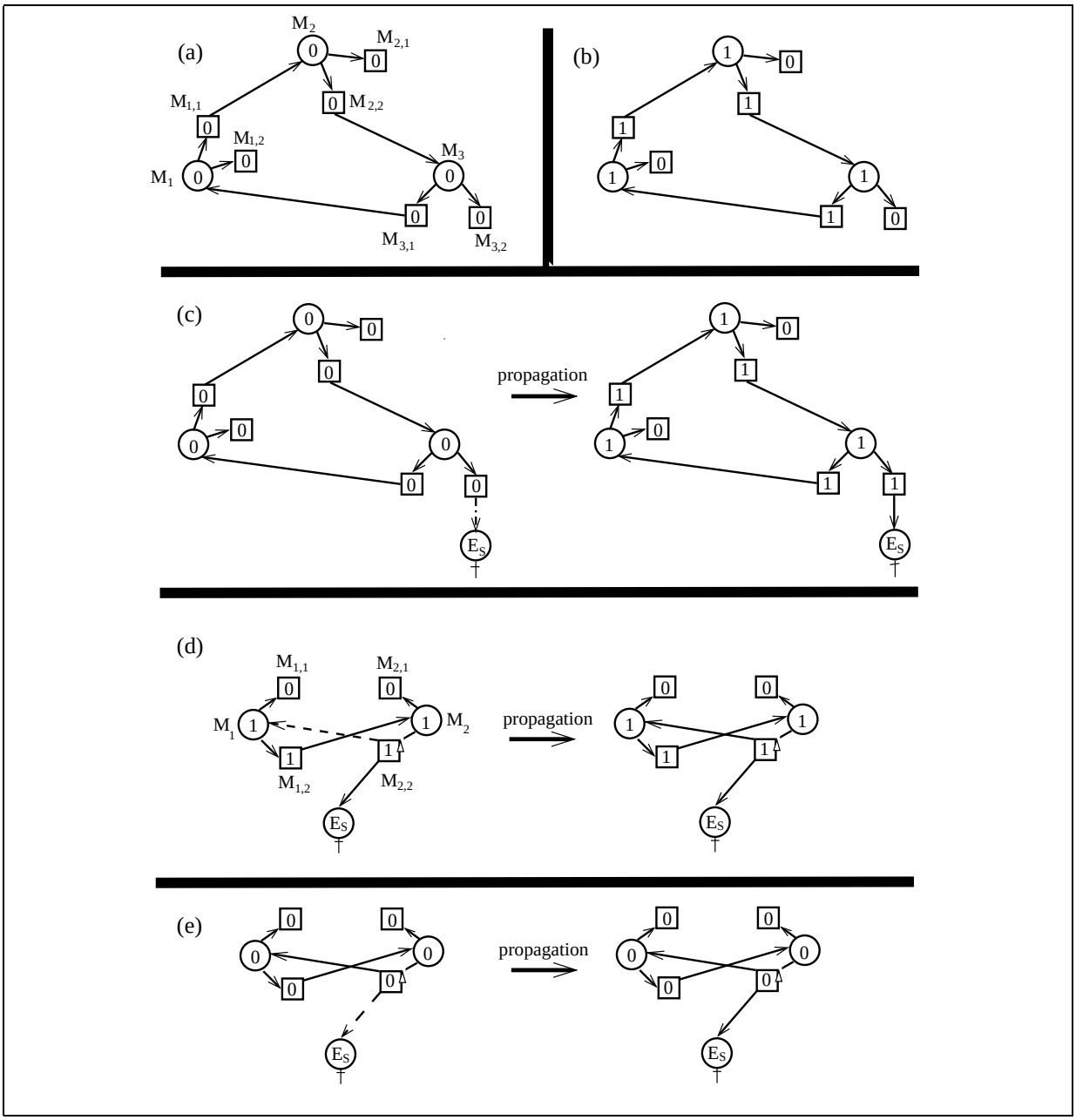


FIG. 2.4 – Exemples de cas de multiplicité des solutions au problème de marquage

que :

- soit il existe un $M_{i_k} \in M$ tel que $M_{i_j} = M_{i_k}$ (cas 3)
- soit il existe un $M_{i_k} \in M$ tel que $M_{i_j} = \max\{0, M_{i_k}\}$ (cas 4)

Le deuxième point correspond au cas évoqué par les graphes (a) et (b). Il offre *a priori* deux solutions pour M_{i_k} : 0 ou 1. Pour le premier point, M_{i_k} peut valoir *a priori* 0, 1 ou -1. Mais, tous ces cas sont-ils possibles ?

En premier lieu, il faut remarquer que l'utilisation des équations ne permet de déduire qu'un marquage vaut 0 uniquement lorsque l'état associé ne possède pas de transition vers un autre état. En effet, la résolution de proche en proche s'effectue à partir des états terminaux E_E et E_S, ce qui permet de fixer les valeurs des marquages à 1 ou -1. On en déduit que s'il existe des marquages valant 0 dont l'état e_i associé possède au moins une transition vers un autre état e_j, alors ces marquages sont dans M. Mais, cela signifie-t-il que tous les marquages de M sont nuls ?

Prenons un état e_i dont le marquage est dans M. e_i ne peut pas avoir de transitions vers un e_j extérieur à M, de marquage connu $M_j = 1$. En effet, dans ce cas, la valeur de M_i serait alors connue et fixée à 1 (cas 1). Or, si un marquage M_i de M vaut 1, cela signifie que cette valeur a été propagée par une transition aboutissant à l'état associé e_i (puisque la valeur initiale de M_i est 0). D'après la remarque précédente, cette transition ne peut provenir que d'un élément de M. Ce qui reporte le problème à un autre M_i de M, d'où une non-résolution du problème dans l'hypothèse d'un marquage de M valant 1. Cela montre l'impossibilité que les marquages de M puissent être égaux à 1. Émettons l'hypothèse qu'un marquage M_i de M soit égal à -1. Dans ce cas, il est régi par le cas 3 (le cas 4 est impossible). Le passage du marquage de 0 à -1 n'est possible que si toutes les transitions issues de E_i possèdent un marquage -1. Comme M_i est dans M, il existe forcément une transition menant à un marquage de M. Ce marquage doit donc avoir été mis à -1 précédemment. Comme dans le cas précédent, le problème du passage de la valeur de M_i de 0 à -1 est reporté sur un autre élément de M. Cela montre également l'impossibilité de notre hypothèse M_i = -1. Par conséquent, l'intégralité des marquages de M ont une valeur 0.

Donc, nous venons de montrer l'unicité de la valeur des marquages obtenus par l'algorithme de propagation, dans le cas où le système vérifie la propriété (P_ϵ).

Considérons à présent le cas particulier où il n'existe pas de transition vers l'état E_S : il correspond à un problème de viabilité. Nous allons montrer le même résultat que précédemment, mais sans utiliser la propriété (P_ϵ). Les modifications au niveau des équations sont peu nombreuses : le cas 4 n'existe plus, puisque le fait qu'il n'existe pas de transition à partir d'un $e_{i,k}$ suffit pour montrer que le marquage de e_i vaut 0. Le cas d'indétermination est donc uniquement le cas 3. D'autre part, la propriété (P_ϵ) n'est utilisée précédemment que pour éliminer la possibilité d'une transition à partir d'un état associé à M jusque vers un état de marquage 1 n'appartenant pas à M (c'est précisément le cas montré par les graphes (d) et (e), pour lequel l'ordre d'apparition des transitions change le marquage final). Dans notre cas, ce problème n'existe plus. En effet, l'indétermination sur M_i existe s'il existe au moins une transition partant $e_{i,k}$ vers un état e_j associé à M, sans qu'il existe une transition de $e_{i,k}$ vers un état non associé à M (dans le cas contraire, le minimum des deux marquages serait égal à -1, quel que soit M_j). L'hypothèse $M_i = -1$ implique donc que l'état e_i soit passé d'un marquage 0 à un marquage -1. Pour cela, l'unique possibilité est que toutes les transitions partant de e_i aboutissent à des états marqués à -1. Or, nous venons de montrer qu'il existe au moins une transition allant de e_i vers un état associé à M. Il faudrait

donc que le marquage de celui-ci soit passé préalablement de 0 à -1. Ce qui reporte le problème à un autre état associé à M. D'où la conclusion que les marquages indéterminés par résolution de proche en proche sont égaux à 0. Ce qui termine la démonstration du troisième point de la proposition 1.

2.4.4 Signification de la valeur des marquages

Une particularité de l'algorithme CbL est que la valeur du marquage d'un état e_i est directement liée à la fiabilité de la politique de commande empruntant e_i. À ce sujet, voici trois propositions intéressantes :

Proposition 2 *lorsque le système respecte (P_ϵ), le marquage M_i d'un état e_i vaut 1 dans l'unique cas où il existe au moins une politique de commande empruntant e_i parvenant à coup sûr à l'objectif E_S.*

Proposition 3 *lorsque le problème de choix de commande est markovien, le marquage M_i d'un état e_i vaut -1 dans le cas où aucune politique de commande empruntant e_i ne peut éviter E_E.*

Proposition 4 *Pour un problème de viabilité, si on constate à un moment de l'apprentissage qu'il existe des états formant un cycle ou plusieurs cycles, dont les marquages possèdent une valeur 0, cela signifie que la politique de commande associée au parcours de ce ou ces cycles n'a jamais échoué jusqu'à ce moment précis.*

Pour prouver la proposition 2, il suffit de montrer qu'il existe un chemin de l'état e_i vers l'état E_S, déterminé uniquement par la séquence des états transitoires choisis (c'est-à-dire des actions sélectionnées). Dans le cas où le système respecte (P_ϵ), nous avons montré que la résolution de proche en proche du système d'équations à partir des états liés à E_S détermine l'intégralité des marquages à 1. Or, la résolution de proche en proche emprunte les transitions existant dans le graphe entre les états $e_{i,k}$ et les états e_j. Donc, pour chaque état possédant un marquage 1, il existe un chemin vers l'état E_S. Ce chemin est déterminé par la connaissance des commandes appliquées car l'hypothèse (P_ϵ) indique qu'un état e_i et une action a_k conditionnent l'état résultant e_j. Par conséquent, si on connaît l'état e_i, une séquence de commandes détermine précisément le parcours dans le graphe d'états. Ce qui montre la proposition 2.

Pour montrer la proposition 3, nous allons d'abord constater que, s'il existe au moins un état e_i de marquage -1, alors il existe un e_j pour lequel il existe une probabilité non nulle que l'état suivant soit E_E. En effet, le marquage -1 est obtenu forcément par la résolution de proche en proche du système d'équations (d'après la proposition 1) ; elle ne peut s'amorcer à partir de l'état E_E que s'il existe un marquage M_j tel que l'ensemble des $M_{j,k}$ soient égaux à -1. Et cela n'est possible que si, quelle que soit la commande choisie, il existe une transition vers E_E. Par conséquent, lorsque le système se situe dans l'état e_j, la probabilité p_j pour qu'il arrive en E_E est non nulle. Et, grâce à l'hypothèse markovienne, p_j ne dépend que de la présence du système dans

l'état e_j. Ce dernier joue alors le même rôle que l'état E_E. Si e_i est différent de e_j, on montre de même qu'il existe un état e_l dont la probabilité de transition p_l vers $E_j \bigcup E_E$ est non nulle, quelle que soit la commande choisie en e_l. Or, l'aboutissement à e_j conduit avec une probabilité p_j vers E_E. Cela implique qu'il existe une probabilité non nulle $p_l.p_j$ d'atteindre E_E à partir de e_l. Le même raisonnement est appliqué jusqu'à ce que e_i fasse partie de l'ensemble des états pour lesquels la probabilité d'atteindre E_S est non nulle. Cela se termine forcément puisque le nombre d'états du système est fini.

La proposition 4 se montre utilisant le fait qu'à partir d'un état e_i de marquage 0 faisant partie d'un cycle, il existe une commande pour laquelle les transitions associées aboutissent toutes à des états de marquage 0 (dans le cas contraire, le marquage de e_i serait à -1).

2.4.5 Exploration de l'espace d'états

La sous-section précédente a permis de montrer que la valeur du marquage associé à un état est discriminante dans le cadre du choix d'une commande fiable (sous les contraintes auxquelles la proposition 1 est associée). Le choix d'une action associée à un marquage 1 sera préféré à tous les autres car, d'après la proposition 2, cette action permet d'arriver à un état menant à l'objectif à coup sûr. À un degré moindre, une action de marquage 0 sera choisie car, d'après la proposition 4, elle fait partie d'un cycle d'états viables ou elle n'a jamais été explorée. Toutes les commandes possédant le même marquage sont donc, d'une certaine manière, équivalentes du point de vue de la fiabilité de la politique de commande dans lesquelles elles s'inscrivent.

Comment le système explore-t-il l'espace d'états à partir de la connaissance des marquages? L'exploration a lieu totalement au hasard lorsque tous les marquages associés aux $e_{i,k}$ sont égaux à 0. La découverte d'une transition vers l'état E_E condamne certaines commandes en fixant des marquages à -1, alors que la découverte d'une transition vers l'état E_S favorise d'autres commandes en fixant des marquages à 1. On se rend compte que, pour un état e_i donné, lorsque la discrimination entre les commandes est effectuée, le choix se porte vers le même ensemble de commandes (ayant le marquage le plus élevé). Ce système revient à exécuter répétitivement la même séquence de commandes, jusqu'à ce que l'expérience montre un défaut dans la fiabilité de la politique de commande[1] (ce qui aboutira à un changement du marquage associé à une partie des commandes, donc à un changement de la politique de commande). Une séquence est donc abandonnée si un unique cas d'échec est détecté. Et, si aucun défaut de fiabilité n'est constaté par l'expérience, il est probable que la politique de commande engendrée ne soit pas optimale, car deux commandes de marquage identique ne peuvent pas être discriminées.

Imaginons que l'algorithme ne trouve pas de politique de commande fiable. Est-on certain d'avoir exploré l'ensemble des séquences susceptibles de former une politique fiable? La proposition suivante répond à cette question par l'affirmative :

Proposition 5 *Le mécanisme d'exploration induit par l'algorithme de choix de commande conduit à une exploration exhaustive de l'espace d'états jusqu'à ce qu'une politique fiable soit trouvée.*

Admettons qu'il existe une politique de commande fiable. D'après les propositions 2, 3 et 4

1. La signification de la fiabilité d'une politique de commande dépend de la catégorie de problème fixée : atteinte d'objectif (fiabilité = atteinte à coup sûr de l'objectif) ou problème de viabilité (le système ne sort jamais de sa zone de viabilité)

établissant une correspondance entre fiabilité et valeur des marquages, cela signifie qu'il existe un ensemble d'états e_i et $e_{i,k}$, de valeur de marquage maximale (1 pour un problème d'atteinte d'objectif et 0 pour un problème de viabilité). Pour un problème de viabilité, la proposition 3 indique que les états/actions faisant partie d'une politique fiable ne peuvent en aucun cas posséder un marquage -1, à aucun moment de l'apprentissage : ces marquages restent donc constamment à 0. D'autre part, l'élimination de possibilités se fait par changement du marquage de 0 (marquage initial) à -1 : la possibilité n'est plus jamais réessayée. Comme on sait qu'il reste obligatoirement des marquages à 0, cette élimination est limitée et conduit à la découverte d'une politique de commande fiable. Pour un problème d'atteinte d'objectif, si on sait qu'il existe une politique fiable, cela signifie avant tout qu'il est possible d'atteindre l'état E_S. Lorsque l'état E_S n'est pas atteint, les valeurs des marquages ne sont pas discriminées et une exploration aléatoire de l'espace d'états est menée. Celle-ci est exhaustive et aboutit forcément à E_S (puisqu'il existe une possibilité d'atteindre E_S). La propagation de la valeur 1 aux marquages associés aux états/actions menant à l'objectif est alors effectuée. On connaît alors une politique fiable d'atteinte d'objectif (par hypothèse (P_e), une commande exécutée à partir d'un état e_i aboutit à un unique état e_j.

2.4.6 Convergence et prédictibilité de l'algorithme CbL

Le problème de la convergence de l'algorithme d'apprentissage est facilement réglé. En effet, seul l'ajout d'une transition dans le graphe d'états peut modifier la valeur des marquages. Si le nombre d'états est fixé, le nombre de transitions est forcément fini et, lorsque chacune des transitions possibles a été explorée, les marquages n'évoluent plus. Nous pouvons même préciser que la convergence a lieu en temps fini : ce temps est majoré par le temps qu'il faudrait pour visiter l'ensemble des états atteignables du système en exécutant une action au hasard, à chaque pas de temps.

La réelle difficulté est de connaître la nature du résultat de l'apprentissage. La proposition 1 donne l'existence d'un lien fonctionnel entre un graphe d'états déterminé (états de type e_i et $e_{i,k}$, transitions et états terminaux E_E et E_S) et le marquage associé aux états (sous les hypothèses restrictives évoquées pour l'obtention d'un ensemble unique de marquages). Ainsi, si deux apprentissages distincts conduisent à deux graphes d'états identiques, la proposition 1 prouve que les marquages obtenus sont également identiques. La sous-section précédente précise que, lorsque l'algorithme d'apprentissage trouve une politique de commande dont l'expérience n'a pas mis en défaut la fiabilité, celle-ci est invariablement répétée. Or, il peut exister plusieurs politiques fiables et la découverte d'une d'entre-elles dépend du tirage aléatoire entre deux commandes de même marquage. Dans ce contexte, il faut abandonner l'idée que l'algorithme peut converger vers une politique de commande unique. Par contre, on peut montrer la proposition suivante :

Proposition 6 *Pour un ensemble d'états donné, il existe un ensemble d'associations état/commande formant une politique de commande fiable (en terme d'objectif ou de viabilité) si et seulement si l'algorithme d'apprentissage CbL découvre une politique de commande fiable. Ce résultat est valable dans la mesure où les contraintes associées à la proposition 1 sont satisfaites.*

La preuve de cette proposition est directement déduite de la proposition 5. Ce résultat est intéressant car il rend l'algorithme CbL prédictible, dans le sens où on sait que si aucune politique fiable n'est découverte, alors il n'en existe pas, du moins en utilisant l'ensemble d'états spécifié

a priori. Le fait qu'il existe une politique fiable et qu'aucune n'est trouvée grâce à CbL signifie que les contraintes d'utilisation de l'algorithme ne sont pas satisfaites.

2.4.7 Utilisation de l'algorithme CbL(α)

L'algorithme CbL(1) possède un défaut important, dont nous montrerons un exemple dans la section suivante. Un état est marqué à 1 s'il existe une politique de commande fiable passant par celui-ci. Or, pour le problème du labyrinthe (voir la section suivante), beaucoup d'états du système sont inclus dans une politique fiable menant à l'objectif. Cela signifie que beaucoup d'états ont un marquage 1. En particulier, pour un même état, plusieurs actions peuvent mener à un état de marquage 1. Dans ces conditions, quelle action choisir ? L'algorithme de choix de l'action indique qu'il faut en choisir une au hasard. On comprend dans ces conditions que le marquage à 1 n'est pas très informatif parce qu'il ne permet pas de diriger le système vers l'objectif.

L'algorithme CbL(α) résout en partie cette difficulté en offrant un ensemble fini de marquages intermédiaires entre le marquage 0 et le marquage 1. L'invariant structurel force alors deux états de marquage strictement positif, reliés par une transition, à posséder deux marquages distincts. On peut alors « remonter » vers l'objectif grâce à l'algorithme de choix d'action, en évitant l'utilisation permanente du hasard.

D'un point de vue théorique, l'algorithme CbL(α) se comporte de la même manière que CbL(1). Des résultats comparables peuvent être montrés pour des problèmes d'atteinte d'objectif. Pour des problèmes de viabilité, les deux algorithmes sont identiques. La valeur de α, pour $\alpha < 1$, n'influence aucunement le résultat de l'algorithme, c'est-à-dire la politique de commande obtenue.

2.5 Propriété d'incrémentalité de l'algorithme CbL

2.5.1 Qu'entendons-nous par « incrémentalité » ?

Prenons l'exemple du problème du labyrinthe. Imaginons qu'une politique de commande ait été découverte, menant à un objectif situé dans une certaine case du labyrinthe. Que se passe-t-il si la cible est déplacée dans une autre case? Que se passe-t-il si la structure du labyrinthe est modifiée (construction de nouveaux murs, mise en place de nouvelles ouvertures) ? Nous souhaiterions que le système puisse s'adapter à ces changements pour atteindre à nouveau la cible. S'il en était capable, il ferait preuve de capacités d'incrémentalité.

Nous allons montrer dans cette section qu'on peut se servir des propriétés établies dans la section précédente pour garantir l'incrémentalité de l'algorithme CbL(α). Cela est ensuite illustré dans la section applicative 2.6.

2.5.2 Découverte de la suppression d'une cible

Nous savons que, dès que le système se trouve dans un état de marquage strictement positif, le mécanisme de choix d'action lui permet de se guider à coup sûr vers un objectif (rattaché à l'état E_S). Cela n'est vrai que si le contexte respecte la propriété (P_ϵ). Nous supposons que cette propriété est satisfaite. Dans ce cadre, si un état $e_{i,k}$ du système possède une transition vers E_S, nous savons que cette transition est unique. Donc, si une nouvelle transition vers un état différent de E_S est découverte à partir de $e_{i,k}$, c'est que la cible n'existe plus ou qu'elle est

accessible à partir d'un autre état. Nous éliminons alors l'ancienne transition de $e_{i,k}$ vers E_S, qui provoque, par réaction, un changement des marquages du système. S'il n'existe plus de connexion à l'état terminal E_S à la suite de la suppression de la transition, nous savons que l'ensemble des marquages qui étaient strictement positifs deviendront égaux à 0. S'il existe plusieurs états liés à E_S, l'ensemble des états appartenant à une politique de commande empruntant la transition supprimée pourront être modifiés. Deux cas se présentent pour le marquage associé à un état e_i compris dans un chemin passant par la transition supprimée :

- il existe un autre chemin entre e_i et E_S : dans ce cas, nous savons que le marquage de e_i restera strictement positif (mais pourra changer de valeur).
- il n'existe pas d'autre chemin entre e_i et E_S : dans ce cas, nous savons que le marquage de e_i sera égal à 0.

La prise en compte de la suppression de la cible se fait à l'instant de la suppression de la transition.

2.5.3 Découverte de l'ajout d'une cible

Nous nous trouvons dans un cas identique à celui de la sous-section précédente. Imaginons qu'à partir d'un état $e_{i,k}$ nous découvrions une nouvelle transition vers E_S alors qu'une transition vers un autre état existe déjà. Le respect du contexte (P_ϵ) impose qu'il y ait eu une modification de l'environnement se traduisant à la fois par l'ajout de cette transition vers E_S et la suppression de l'ancienne. En réaction à ce changement, une modification des marquages est possible :

- les états pour lesquels un chemin vers E_S est accessible grâce à l'ajout de la transition ont un marquage modifié à 1.
- les états empruntés par une politique de commande, passant par la transition supprimée et aboutissant à E_S ont un marquage toujours strictement positif, mais la valeur de ceux-ci peut être modifiée.

2.5.4 Découverte de la suppression d'un obstacle

L'effet de la suppression d'une source d'erreur dans l'environnement ne change rien à la politique de commande du système puisque celle-ci est censée ne plus visiter les états susceptibles d'être connectés à l'état terminal E_E. Il n'y a donc aucune raison pour que le système découvre cette modification de son environnement : il lui faudrait exécuter un ensemble de commandes menant habituellement à l'échec pour découvrir ce changement, et l'algorithme de choix de la commande ne le permet pas.

La découverte de la suppression d'une source d'erreurs est utile uniquement dans un cas précis. Prenons l'exemple du labyrinthe pour illustrer ce cas. Supposons que le système ne puisse rejoindre la cible, parce qu'il est entouré d'obstacles. Si un de ceux-ci est retiré à un moment donné, il lui est alors possible d'aller à l'objectif. Dans ce cas, on peut imaginer une stratégie simple pour découvrir un changement de cette nature. Il suffit de supprimer régulièrement l'ensemble des transitions avec l'état terminal E_E, ce qui revient à autoriser l'exécution de commandes qui étaient associées auparavant avec l'état E_E (se cogner dans un obstacle). Cette modification permet une nouvelle exploration des états, qui aboutira à coup sûr à la découverte du passage menant à l'objectif.

2.5.5 Découverte de l'ajout d'un obstacle

Cette découverte se fait naturellement lorsqu'une transition vers l'état E_E est créée à partir d'un état $e_{i,k}$ possédant déjà une transition vers un autre état. En se fiant au respect de la contrainte (P_ϵ), il suffit de supprimer l'ancienne transition et de rajouter celle menant à E_E.

2.5.6 Conclusion : lien entre la capacité d'incrémentalité de l'algorithme CbL et l'invariant structurel engendré par (P_ϵ)

Pour obtenir le caractère d'incrémentalité de l'algorithme CbL(α), il nous faut supposer que le contexte de l'apprentissage respecte la propriété (P_ϵ). En effet, nous nous basons sur ce postulat pour discerner une modification de l'environnement (cas où une deuxième transition doit être créée à partir d'un état $e_{i,k}$).

La propriété (P_ϵ) peut être vue comme un invariant structurel du graphe, à l'instar de la contrainte d'équilibre reliant les marquages d'états voisins. Si on considère que cette propriété doit être respectée à chaque pas de temps, la tentative de création d'une deuxième transition à partir d'un état $e_{i,k}$ invalide cet invariant. Pour le rétablir, il suffit de supprimer l'ancienne transition. La modification de l'environnement est alors perçue par le système comme une double opération d'ajout et de suppression de transitions à partir d'un même état $e_{i,k}$. L'observation de ce changement à travers le graphe d'états est alors une propriété émergente de l'invariant structurel sous-jacent à la propriété (P_ϵ).

Ce changement mineur de l'algorithme CbL est effectuée simplement, en modifiant légèrement l'algorithme 2.2, page 46 : au niveau de la création d'une transition, il suffit de regarder si une autre transition partant de $e_{i,k}$ existe déjà et la supprimer le cas échéant. Dans le cas d'une suppression, il faut appeler l'algorithme 2.1, page 45, pour une mise en cohérence des marquages. La politique de commande est alors modifiée naturellement par l'ajout ou la suppression d'une cible (c'est-à-dire par la découverte d'une nouvelle transition vers E_S), ainsi que par l'ajout d'une source d'erreurs (c'est-à-dire par la découverte d'une nouvelle transition vers E_E). La prise en compte de la suppression d'une source d'erreurs n'est intéressante que dans le cas où le système est confiné dans un ensemble restreint d'états viables sans pouvoir atteindre l'objectif. Dans ce cas, il est nécessaire d'ajouter une stratégie de haut niveau, commandant régulièrement la suppression des transitions vers E_E, afin de permettre une nouvelle phase d'exploration, permettant si possible de trouver un chemin vers l'objectif.

2.6 Problème du labyrinthe

2.6.1 Introduction

Dans cette section, nous donnons un exemple applicatif de l'algorithme CbL illustrant l'atteinte d'objectif : le problème du labyrinthe. Nous utiliserons l'algorithme CbL(α).

Voici nos objectifs :

- mettre en évidence les propriétés et les défauts de l'algorithme CbL
- comparer les performances de l'algorithme CbL avec celles du Q-Learning

2.6.2 Position du problème

L'environnement est un rectangle découpé en un ensemble régulier de cases, chacune associée à un état du système. Chaque case est soit vide, soit occupée par un mur ou par une cible. On suppose que le système possède quatre commandes possibles : aller dans la case adjacente de droite, de gauche, en haut ou en bas. De plus, on suppose qu'il connaît exactement l'état dans lequel il se trouve à tout moment. Mais, il ne sait pas *a priori* où sont les obstacles et les cibles (les bords du labyrinthe sont considérés comme des obstacles, dans lesquels le système peut se cogner). L'objectif est d'apprendre à atteindre une des cibles présentes dans le « labyrinthe » en utilisant les quatre commandes de base. Le contexte d'apprentissage est un cas typique pour lequel (P_ϵ) est respecté.

Voici comment nous allons représenter le graphe d'états du système. La figure 2.5 résume l'ensemble des types de graphes que nous allons utiliser. Les graphes (a) et (b) concernent la politique de commande générée à partir du graphe d'états. Les cases sont représentées par les lignes en trait fin. Prenons le graphe (a). La direction des flèche indique l'action à entreprendre. Celle-ci correspond à l'état transitoire possédant le marquage le plus élevé (et strictement positif). Lorsque le marquage maximum est inférieur ou égal à 0, aucune flèche n'est indiquée. Lorsque plusieurs flèches existent dans une même case, cela signifie que plusieurs actions possèdent le même marquage maximum. Le graphe (b) donne, pour le même exemple, la valeur précise du marquage des états. Le graphe (c) indique l'ensemble des transitions créées dans le graphe d'états : pour chaque état, des flèches partant du centre de la case correspondant à celui-ci indiquent l'existence d'une transition vers la case du haut, du bas, de gauche ou de droite.

2.6.3 Protocole d'apprentissage

Les termes *essai* et *itération* ont une signification similaire à celle définie dans le chapitre 1. Le lecteur pourra s'y référer.

Les performances de l'algorithme CbL seront mesurées avec les indicateurs suivants :

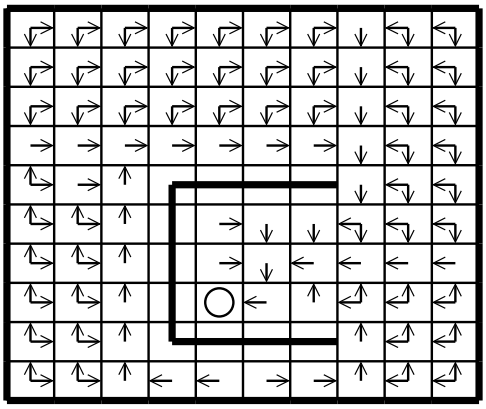
- pour s'assurer de la convergence de l'algorithme CbL, nous évaluerons à chaque essai d'apprentissage le nombre d'états dont le marquage a été modifié [1].
- l'évolution du nombre moyen d'échecs en fonction du numéro de l'essai. Ce nombre moyen est obtenu en considérant l'ensemble des apprentissages effectués
- le nombre moyen d'itération, pour chaque essai, avant l'atteinte de l'objectif (si l'objectif n'est pas atteint ou si un obstacle est rencontré, le nombre maximum d'itérations est considéré).

Voici le protocole d'apprentissage. 1000 apprentissages successifs utilisant l'algorithme CbL ou la méthode du Q-Learning seront effectués. Chaque apprentissage comporte 5000 essais. Le nombre maximum d'itération par essai est fixé à 1000. Si le système n'a pas trouvé la cible ou ne s'est pas cogné avant 1000 itérations, l'essai est terminé. L'initialisation de l'état du système au début de chaque essai se fera aléatoirement sur une case « libre ».

Le paramètre α de l'algorithme CbL sera fixé à 0.99 . Comme nous l'avons déjà mentionné, ce paramètre n'influe pas sur le résultat de l'apprentissage : l'unique condition est qu'il soit compris dans l'intervalle]0,1[. D'autre part, nous avons fixé le paramètre d à 100 (le paramètre d indique

1. Pour la méthode du Q-Learning on sommera les valeurs absolues des modifications apportées aux marquages des états pour chaque essai.

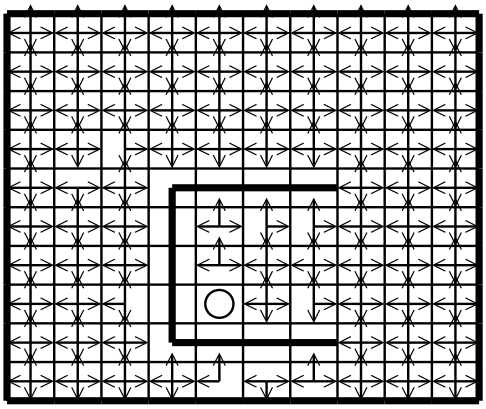
(a)



(b)

0.835	0.843	0.851	0.860	0.869	0.878	0.886	0.895	0.886	0.878
0.843	0.851	0.860	0.869	0.878	0.886	0.895	0.904	0.895	0.886
0.851	0.860	0.869	0.878	0.886	0.895	0.904	0.914	0.904	0.895
0.860	0.869	0.878	0.886	0.895	0.904	0.914	0.923	0.914	0.904
0.851	0.860	0.869					0.932	0.923	0.914
0.843	0.851	0.860		0.951	0.961	0.951	0.941	0.932	0.923
0.835	0.843	0.851		0.961	0.970	0.961	0.951	0.941	0.932
0.826	0.835	0.843			0.980	0.951	0.941	0.932	0.923
0.818	0.826	0.835					0.932	0.923	0.914
0.810	0.818	0.826	0.818	0.810	0.904	0.914	0.923	0.914	0.904

(c)



Commentaires :
Dans l'ensemble des graphes, les traits fins correspondent à la limitation des cases du labyrinthe, alors que les traits épais signifient la présence d'un mur ou d'une bordure du labyrinthe. La ou les cibles sont représentées par un cercle. Chaque case représente un état du système.
Le graphe (a) montre un exemple de politique de commande établie après un apprentissage. Les flèches centrées sur les cases du labyrinthe indiquent les commandes associées à un marquage maximum strictement positif. Dans le cas où une case ne possède pas de flèche, cela signifie que le marquage associé à chacune des commandes issues de cette case est inférieur ou égal à 0. Lorsque plusieurs flèches sont issues du centre d'une case, plusieurs commandes possèdent un marquage maximum, donc plusieurs commandes sont susceptibles d'être exécutées par l'algorithme de choix de la commande.
Le graphe (b) est issu du même apprentissage que le graphe (a). Il montre, pour chacun des états, la valeur du marquage qui lui est associé.
Le graphe (c) montre l'exploration des états du système après l'apprentissage montré par les graphes (a) et (b). Les flèches centrées sur les cases indiquent que les commandes correspondantes ont été essayées au moins une fois.
Les graphes de type (a) ou (c) ayant le même aspect, nous préciserons à chaque fois s'ils représentent la politique de commande ou l'exploration de l'espace d'états.

FIG. 2.5 – Exemples de graphes pour le problème du labyrinthe.

le nombre de marquages distincts strictement supérieurs à 0 : voir la sous-section 2.2.2, page 41). Nous verrons l'influence de ce paramètre dans l'expérience du labyrinthe. Les paramètres internes du Q-Learning sont ceux utilisés dans l'application du pendule inversé dans le chapitre 1 (voir l'annexe A.2.2, page 85).

2.6.4 Résultats

Les résultats expérimentaux concernent les points suivants :

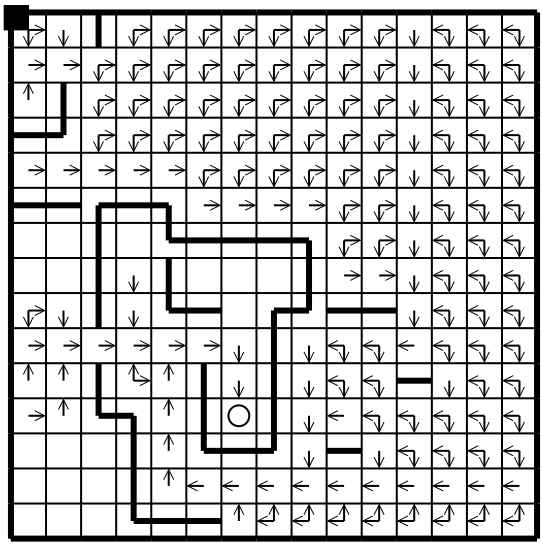
1. les particularités de l'algorithme CbL
2. un comparatif des performances de l'algorithme CbL avec la méthode du Q-Learning avec trace d'éligibilité

Commençons par le point 1. Une première expérience donne un exemple d'apprentissage. Celle-ci consiste à effectuer un unique apprentissage, comportant 1000 essais, en initialisant l'état du système sur la même case, pour chaque essai. L'environnement est celui présenté dans les graphes (a), (b) et (c) de la figure 2.6. L'état initial est indiqué par un carré noir (en haut, à gauche du graphe (a)). Nous remarquons que l'exploration du labyrinthe n'est pas totale (graphe (c)). Cela est dû au fait que, dès que le système va découvrir la cible, l'ensemble des marquages des états explorés jusqu'à ce moment vont être modifiés pour respecter la contrainte d'équilibre. Comme l'ensemble de ces états peuvent conduire vers la cible, ils auront tous un marquage strictement positif. D'après l'algorithme de choix de la commande, les autres états (ayant un marquage nul) ne seront plus explorés à partir de ce moment. Un bénéfice induit par cette modification massive des marquages au moment de la découverte de la cible est que, à partir de ce moment, si on initialise le système dans un état déjà exploré, il conduira vers la cible. Le graphe (a) illustre ce fait : en effet, à partir de n'importe quelle case du labyrinthe, le choix d'une action, symbolisé par une flèche dans le graphe (a), permet au système de se rapprocher de la cible. De plus, dans notre cas particulier, la politique de commande induit un trajet de longueur minimale à partir de n'importe quel état du système (le choix d'une commande a toujours comme résultat de se rapprocher de la cible). Le graphe (d) montre la particularité de l'algorithme CbL : la modification des marquages est massive dès que le système découvre la cible (juste après l'essai 300). Le graphe (e) montre qu'à partir de cet instant, le nombre d'itérations pour aller à la cible est constant et minimal.

Les graphes (a) et (c) de la figure 2.7 illustrent le fait que l'optimalité de la politique de commande dépend de l'exploration des états qui a été effectuée avant la découverte de la cible. Les graphes (a) et (c) donnent les résultats d'apprentissage à partir d'un même environnement, possédant deux cibles. L'état initial est choisi aléatoirement à chaque essai. Le graphe (b) de la figure 2.7 illustre l'exploration des états correspondant à l'apprentissage du graphe (a). Mais le graphe (c) est obtenu à partir d'un graphe d'états initial (au début de l'apprentissage) possédant l'intégralité des transitions possibles (à l'exclusion des transitions menant aux états terminaux). Nous remarquons que la politique de commande pour (a) n'est pas optimale, alors qu'elle l'est pour (c). Cela est dû au manque d'exploration (montré par le graphe (b)).

L'algorithme CbL assure que si un état possède un marquage strictement positif, alors l'algorithme de choix de la commande permet d'aboutir à coup sûr à au moins une cible. Ainsi, **les cibles peuvent être vues comme des points d'attraction**. Chaque état possédant un marquage strictement positif peut être classifié en fonction de la ou des cibles vers lesquelles l'algorithme de choix de commande « attire » le système à partir de cet état. Nous pouvons ainsi

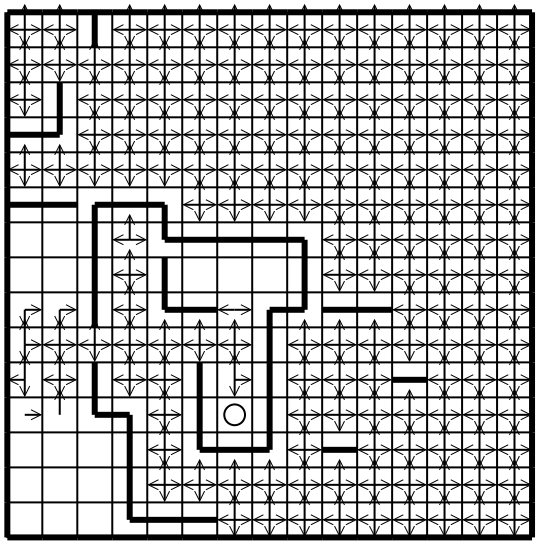
(a) Politique de commande



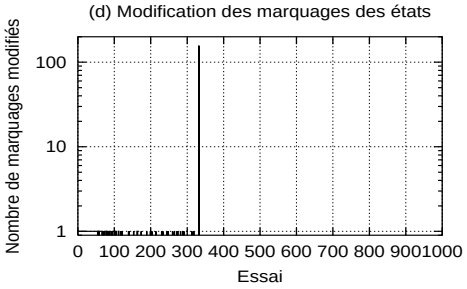
(b) Valeur des marquages des états

0.669	0.676		0.689	0.696	0.703	0.711	0.718	0.725	0.732	0.740	0.747	0.740	0.732	0.725
0.676	0.683	0.689	0.696	0.703	0.711	0.718	0.725	0.732	0.740	0.747	0.755	0.747	0.740	0.732
0.669		0.696	0.703	0.711	0.718	0.725	0.732	0.740	0.747	0.755	0.762	0.755	0.747	0.740
		0.703	0.711	0.718	0.725	0.732	0.740	0.747	0.755	0.762	0.770	0.762	0.755	0.747
0.696	0.703	0.711	0.718	0.725	0.732	0.740	0.747	0.755	0.762	0.770	0.778	0.770	0.762	0.755
					0.740	0.747	0.755	0.762	0.770	0.778	0.786	0.778	0.770	0.762
0.000	0.000		0.000						0.778	0.786	0.794	0.786	0.778	0.770
0.000	0.000		0.923		0.000	0.000	0.000		0.786	0.794	0.802	0.794	0.786	0.778
0.904	0.914		0.932			0.000					0.810	0.802	0.794	0.786
0.914	0.923	0.932	0.941	0.951	0.961	0.970		0.843	0.835	0.826	0.818	0.810	0.802	0.794
0.904	0.914		0.932	0.941		0.980		0.851	0.843	0.835		0.818	0.810	0.802
0.895	0.904			0.932				0.860	0.851	0.843	0.835	0.826	0.818	0.810
0.000	0.000	0.000		0.923				0.869		0.851	0.843	0.835	0.826	0.818
0.000	0.000	0.000		0.914	0.904	0.895	0.886	0.878	0.869	0.860	0.851	0.843	0.835	0.826
0.000	0.000	0.000				0.886	0.878	0.869	0.860	0.851	0.843	0.835	0.826	0.818

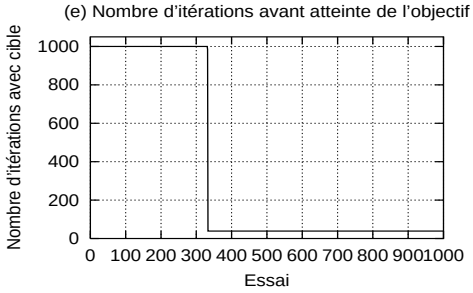
(c) Exploration des états



(d) Modification des marquages des états



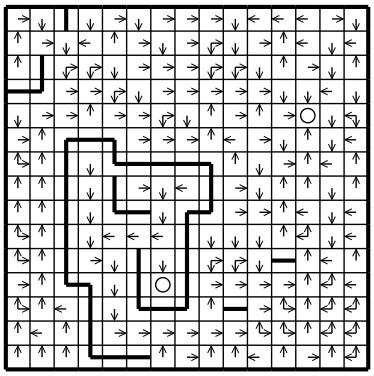
(e) Nombre d'itérations avant atteinte de l'objectif



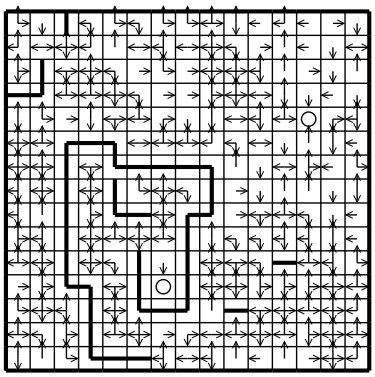
Commentaires : Les graphes (a), (b) et (c) sont obtenus après le même apprentissage, dans un environnement possédant une unique cible (représentée par un cercle). Le carré en haut à gauche du graphe (a) indique l'état d'initialisation du système au début de chaque essai d'apprentissage. Le graphe (a) indique la politique de commande obtenue. Les cases ne possédant pas de flèche n'ont jamais été explorées, ce qui peut être retrouvé sur le graphe d'exploration (c) et sur le graphe (b), montrant des marquages de valeur nulle pour les états non explorés. Sur le graphe (a), si on choisit une case et qu'on suit de proche en proche une des flèches indiquant les commandes possibles, on aboutit de manière certaine à la cible. Cela peut être visualisé dans le graphe (b) par l'existence d'un gradient au niveau des marquages, qui peut être remonté jusqu'à la valeur maximum 1 (sur la cible). Le graphe (d) montre qu'il existe une modification massive des marquages juste après le 300ème essai, ce qui correspond précisément au moment où la cible est découverte (graphe (e)).

FIG. 2.6 – Particularités de l'algorithme CbL (1)

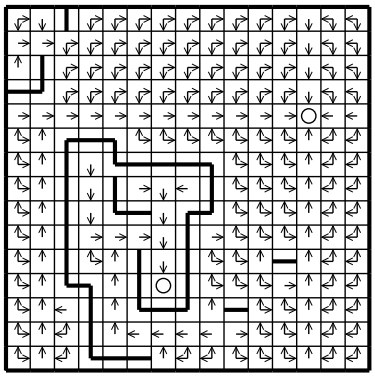
(a) Politique de commande



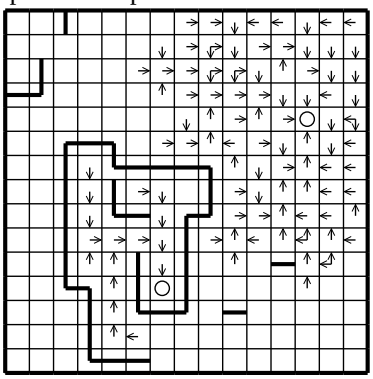
(b) Exploration limitée des états



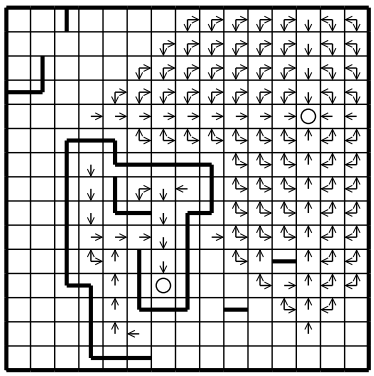
(c) Politique de commande
après exploration totale des états



(d) Politique de commande
lorsque l'ensemble des marquages est petit
après une exploration limitée des états



(e) Politique de commande
lorsque l'ensemble des marquages est petit
après une exploration totale des états



Commentaires :
Les graphes (a), (b) et (c) traitent de l'influence de l'exploration sur la qualité de la politique de commande. Les graphes (a) et (b) montrent la politique de commande issue de deux apprentissages différents. Pour (a), on part, au début de l'apprentissage, d'un graphe d'états ne possédant aucune transition (l'exploration au bout de l'apprentissage est donnée par le graphe (b)), alors que pour (c), on part d'un graphe possédant l'intégralité des transitions réalisables. Le résultat est que le graphe (c) donne une politique de commande optimale (nombre minimal de commandes pour rejoindre une des deux cibles à partir de n'importe quel état). Ce n'est pas le cas pour le graphe (a).
Les graphes (d) et (e) donnent les politiques de commandes obtenues après apprentissage, en fixant la valeur du paramètre d à 10. Cela signifie qu'un état ne pourra avoir un marquage strictement positif que s'il faut moins de 10 commandes à partir de ce dernier pour atteindre l'objectif. Les différences entre (d) et (e) sont les mêmes qu'entre (a) et (c). Dans ces conditions, le graphe (e) montre une politique de commande optimale (tous les états à partir desquels on peut rejoindre une des deux cibles en moins de 10 commandes possèdent un marquage strictement positif, indiqué par la présence d'au moins une flèche). On voit que cela n'est pas vrai pour le graphe (d).

Fig. 2.7 – Particularités de l'algorithme CbL (2)

dégager des zones d'attraction pour les deux cibles. Dans le cas du graphe (a), la zone d'attraction de la cible située dans la partie « sinueuse » du labyrinthe est limitée à une case ; l'ensemble des autres cases explorées sont dans la zone d'attraction de la deuxième cible. Pour le graphe (d), les deux zones sont séparées de manière optimale : à partir de chaque état, on se dirige vers la cible la plus proche, et si un état est à la même distance (en terme de nombre de commandes) des deux cibles, la politique de commande peut amener le système soit vers une cible, soit vers l'autre, d'une manière équiprobable.

Enfin, nous montrons l'influence du paramètre d (fixé à 100 jusqu'ici). Nous rappelons qu'il indique le nombre maximum de marquages strictement positifs autres que 1. Il délimite l'étendue du bassin d'attraction d'une cible, c'est-à-dire l'existence d'un guide vers l'objectif menant à celui-ci au maximum en d itérations. Dans les cas précédents, la valeur de d était largement suffisante. Nous allons revenir à l'exemple précédent, en fixant la valeur de d à 10. Les graphes (d) et (e) sont les mêmes que (a) et (c), mais avec un paramètre d beaucoup plus petit. Les bassins d'attraction sont visibles (les flèches indiquent l'existence d'un marquage strictement positif). On se rend compte que la réduction du bassin d'attraction de la deuxième cible a permis d'augmenter le nombre d'états compris dans celui de la première cible : cela est dû au fait que l'exploration s'est poursuivie autour de la première cible, puisque les états proches de celle-ci sont à plus de 10 cases de l'autre cible (donc ne peuvent pas être marqués positivement à cause d'elle). On remarque que le graphe (d) n'induit pas un bassin d'attraction maximal pour la deuxième cible, pour des raisons de manque d'exploration des transitions possibles. Le graphe (e) montre au contraire que le bassin d'attraction autour des deux cibles comporte l'intégralité des états à partir desquels il faut, au plus, 10 commandes pour atteindre une cible.

En conclusion, ces résultats expérimentaux montrent clairement que **le dilemme exploration/exploitation existe en utilisant l'algorithme CbL** : si on souhaite un résultat optimal, il est nécessaire de dissocier la phase d'exploration (construction du graphe d'états sans prendre en compte les transitions vers les états terminaux) et la phase d'exploitation (construire les transitions vers les états terminaux). Cela signifie qu'on renonce à changer la valeur des marquages dans la phase d'exploration, ce qui peut augmenter considérablement le temps d'apprentissage. À l'inverse, notre algorithme de choix de la commande implique qu'on passe directement à la phase d'exploitation dès qu'une cible est détectée, ce qui entraîne une non optimalité de la politique de commande résultante. Ainsi, l'ensemble des exemples présentés ci-dessus montre l'aspect capital de la topologie du graphe d'états - c'est-à-dire des liens possibles entre les états - pour la détermination d'une politique de commande.

Nous allons, à présent, passer au point 2. Le protocole expérimental indique le mode opératoire. Nous avons utilisé **deux environnements différents**, que nous noterons « environnement 1 » et « environnement 2 » (voir respectivement les figures 2.8, page 66, et 2.9, page 67) : ils se différencient par le nombre d'obstacles et la probabilité d'atteindre une cible en choisissant les commandes au hasard. Commençons par les résultats concernant **l'environnement 1**, qui est le plus simple. L'ensemble des graphes pour cet environnement est donné sur la figure 2.8. Les graphes (a) et (b) montrent les marquages associés à chaque état, respectivement pour CbL et pour la méthode du Q-Learning. Ils sont les résultats du dernier essai du dernier apprentissage. Les graphes (c) et (d) montrent les convergences respectives des l'algorithmes CbL et Q-Learning, alors que le graphe (e) compare les nombres moyens d'itérations pour parvenir à l'objectif et le graphe (f) montre l'évolution comparée du nombre moyen d'erreurs par essai. Nous remarquons

les faits suivants :

- la méthode du Q-Learning permet de trouver une meilleure politique de commande que CbL. En effet, le graphe (e) montre que le nombre moyen d'itérations au bout de 5000 essais est d'environ 8 pour le Q-Learning, alors qu'il est d'environ 11 pour CbL (voir la zone intéressante C sur ce graphe).
- CbL converge plus rapidement que Q-Learning : l'évolution comparée des graphes (c) et (d) montre ce fait. Plus aucune modification de marquage n'est détectée à partir du 200ième essai (zone A) pour CbL, alors que la somme moyenne des modifications des Q-values n'est jamais nulle pour Q-Learning au 5000ième essai (zone B du graphe (d)).
- CbL devient fiable beaucoup plus rapidement que Q-Learning : le graphe (f) montre ce fait. Plus aucune erreur n'est détectée après le 200ième essai pour Cbl (zone D), alors qu'il faut plus de 1000 essais pour obtenir le même résultat avec le Q-Learning.

Le graphe (a) montre la non optimalité de la politique de commande avec CbL, ce qui a été expliqué dans la première série d'expériences. Alors que le graphe (b) montre que la politique de commande est presque optimale après l'apprentissage utilisant le Q-Learning. Elle est due au manque d'exploration après que des chemins vers les cibles aient été trouvés. La convergence rapide est montrée par le graphe (c). Celui-ci met en évidence un pic (traduisant le nombre moyen d'itérations qu'il faut pour découvrir les cibles) qui se situe à la 20ième itération, puis une décroissance très rapide des modifications qui deviennent nulles (en moyenne, donc pour tous les essais) après la 200ième itération. Le graphe (d) donne des résultats identiques mais obtenus au bout d'un nombre d'itérations plus importants, pour la méthode du Q-Learning : le pic se situe vers la 300ième itération et la moyenne des sommes des modifications n'est jamais nul (très faible à la 5000ième itération, mais non nulle).

L'environnement 2, plus complexe, amplifie les résultats que nous venons de donner. L'ensemble des graphes est donné par la figure 2.9. Dans ce cas, la politique de commande pour le QL n'est pas satisfaisante à l'issue des 5000 essais, car l'apprentissage est loin d'être achevé. Nous n'avons pas inclus les données concernant le Q-Learning, car elles ne sont pas représentatives d'un apprentissage achevé ou sur le point de l'être, pour ce nombre d'essais. L'algorithme CbL permet, quant à lui, de trouver une politique de commande satisfaisante (graphe (a)) en un nombre d'itérations limité. Au bout de 200 itérations en moyenne, la cible est atteinte pour la première fois, permettant une modification massive des marquages associés aux états (on relève un pic important sur le graphe (b), suivi d'une chute rapide et d'une annulation du nombre moyen de marquages modifiés (zone A)). Le nombre moyen d'erreurs sur les 1000 essais devient nul après la 200ième itération (graphe (d), zone C) et l'algorithme a convergé dans tous les cas avant la 200ième itération. Le nombre moyens d'itérations pour atteindre l'objectif est constant après la 200ième itération (zone B du graphe (c)). Ces résultats doivent être comparés avec l'absence de résultats satisfaisants pour l'algorithme du Q-Learning au bout du 5000ième essai. Le gain de rapidité de l'apprentissage, qui était d'environ 10 pour l'environnement simple précédent, devient beaucoup plus important pour un environnement plus complexe. D'autre part, on constate (graphe (e)) que l'exploration a été quasiment totale (pour l'exemple d'apprentissage considéré), ce qui signifie que la politique de commande donnée par le graphe (a) est optimale (comme nous l'avons déjà montré au cours des premières expériences).

Nous venons donc de montrer le gain de temps d'apprentissage de l'algorithme CbL par rapport au Q-Learning et la fiabilité de la politique de commande obtenue (il n'y a plus d'erreur au bout d'un certain nombre d'essais). Il nous reste un point très intéressant : tester les capa-

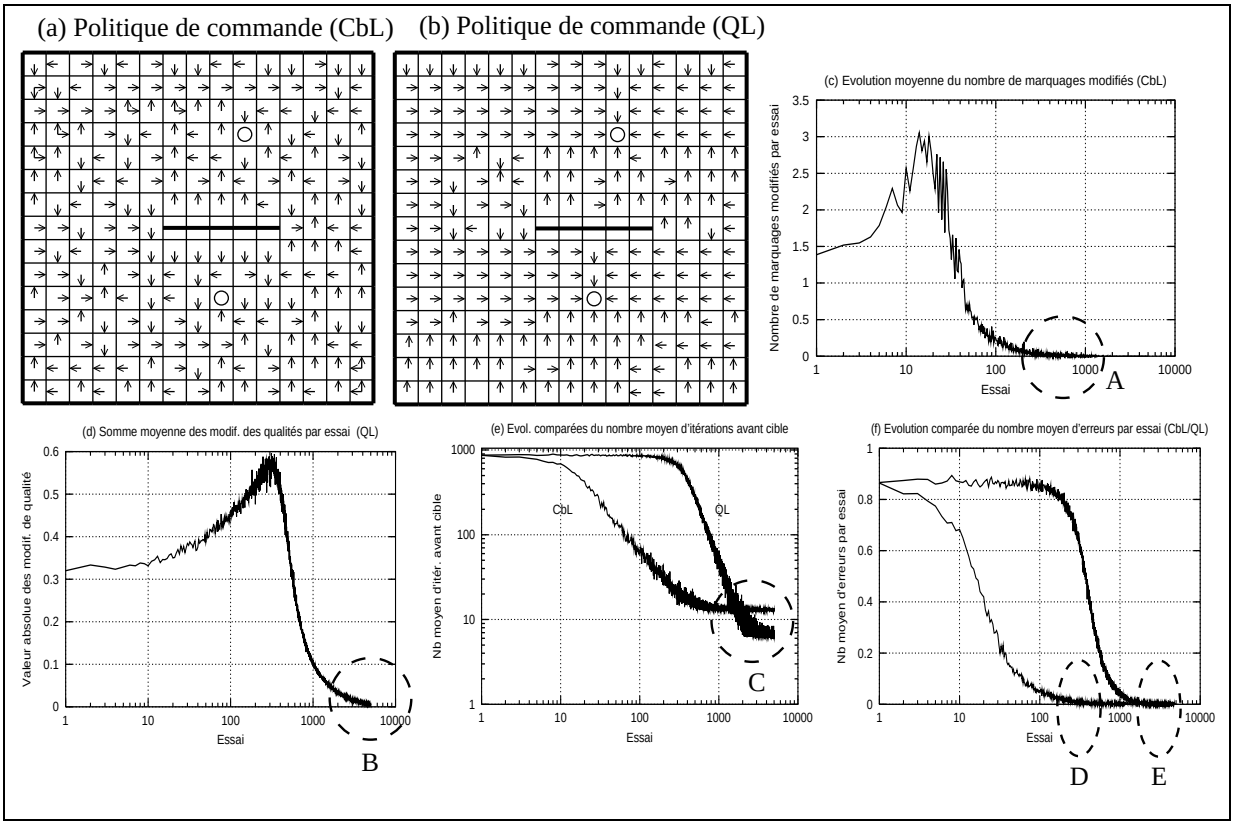


Fig. 2.8 – Résultats pour l'environnement 1

cités d'adaptation du système à une modification de l'environnement. L'ensemble des résultats est donné par la figure 2.10. On confronte le même système, en train d'apprendre par CbL, à un environnement pour lequel on va ajouter successivement une cible ou un mur (la succession des environnements est donnée par les graphes allant de (a) à (f)). Pour chacun des six environnements, on effectue 500 essais d'apprentissage. On passe d'un environnement à l'autre sans réinitialiser les marquages du système. L'apprentissage total comporte donc 3000 essais. Au début du premier essai de **l'unique apprentissage** (donc au début de l'essai 1 pour l'environnement (a)), on suppose que le graphe d'états possède l'ensemble des transitions possibles d'un état à l'autre, en excluant les transitions vers les états terminaux : nous avons vu que c'est une manière d'obtenir une politique de commande optimale après l'apprentissage. Les graphes allant de (a) à (f) indiquent la politique de commande du système à l'issue de l'apprentissage d'un environnement.

L'ensemble des graphes montrent une adaptation parfaite **à l'ajout d'une cible ou d'un mur**, dans la majorité des cas : en effet, la politique de commande à la fin des 500 essais donnés pour chaque ajout d'une cible ou d'un mur est optimale pour les graphes (a),(b),(e),(f). Cependant, on constate que l'ajout des cibles des graphes (c) et (d) ne donne pas une politique de commande optimale : certaines ne sont atteintes à partir d'aucun état du système. Ce phénomène s'explique : lorsque l'ajout d'une nouvelle cible se situe sur un état qui n'est atteint à partir d'aucun autre état (aucune flèche n'arrive sur cet état), il ne s'ensuit aucune modification de la politique de commande. **Dans la partie théorique, ce cas avait été prévu** : le changement du marquage d'un état est dû à une modification dans le marquage des états entre lui et un état terminal. Dans notre exemple, il n'existe pas de transition menant aux cibles et il existe déjà

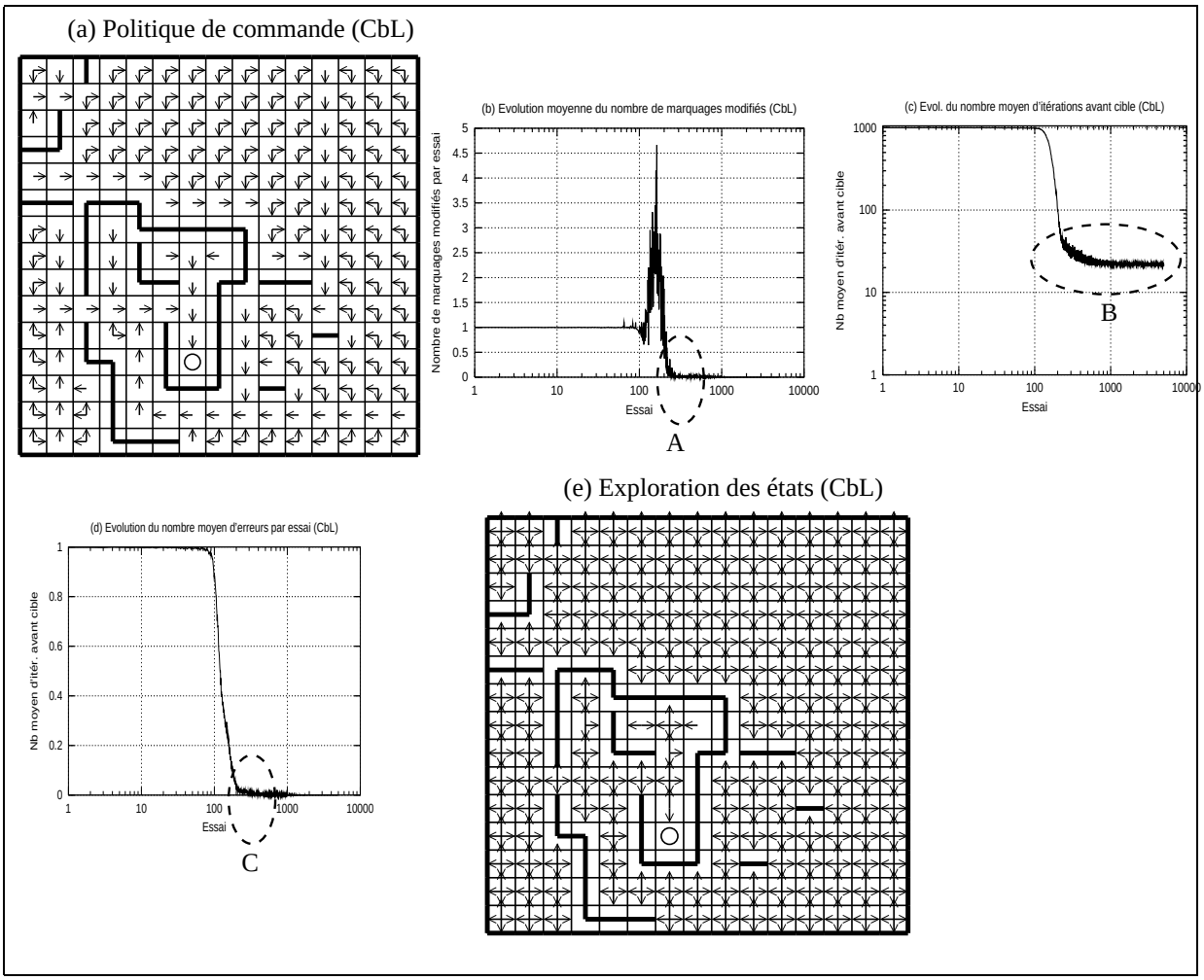
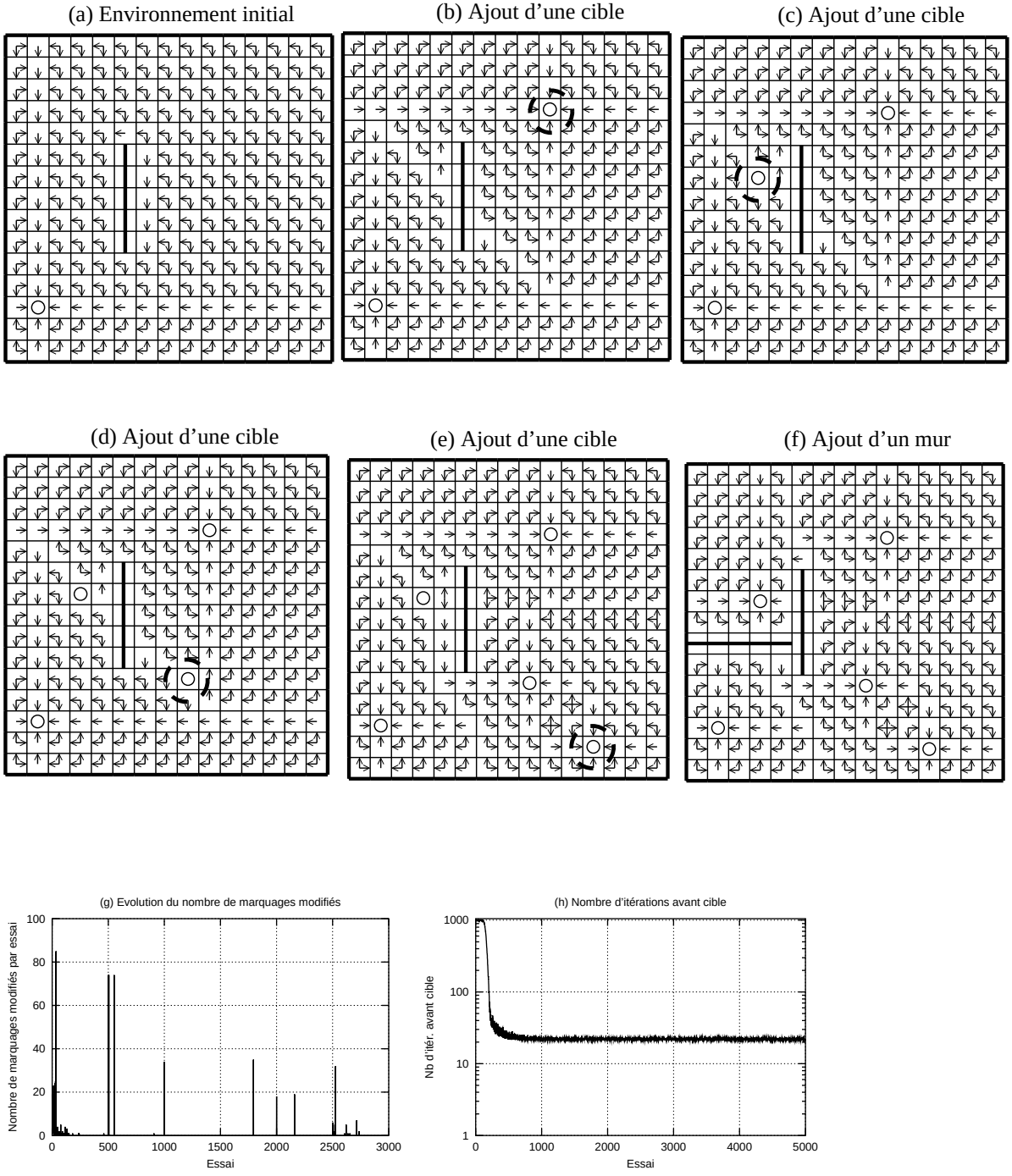


FIG. 2.9 – Résultats pour l'environnement 2



Commentaires :
Les graphes (a) à (f) montrent la politique de commande obtenue après 500 essais d'apprentissage, pour chaque environnement pris successivement. Le graphe (g) montre un phénomène « classique » de l'algorithme CbL : la découverte d'une cible ou d'un mur alors qu'elle n'était pas prévue provoque une modification d'un nombre important de marquages. La majorité des modifications a lieu juste après l'introduction d'un nouvel élément dans l'environnement, ce qui se traduit par des pics autour des itérations 1, 501, 1001, 2001 et 2501.

Fig. 2.10 – Adaptation à des modifications successives de l'environnement

un état terminal. L'ensemble des états pour lesquels ce phénomène se produit sont ceux pour lesquels le marquage est minimum localement (tous les états voisins possèdent un marquage strictement supérieur). Le graphe (e) est intéressant car il montre que le « minimum local » des marquages, sur lequel se trouvent les deux cibles non atteintes, peut être enlevé en rajoutant une nouvelle cible (graphe (e)). Et le graphe (f) indique que l'adaptation a lieu également lorsqu'un nouvel obstacle est ajouté. Enfin, les graphes (g) et (h) indiquent que l'adaptation du système s'effectue en moins de 500 essais pour chaque nouvel environnement. Les pics du graphe (g) sont caractéristiques des modifications massives, limitées dans le temps ; on constate qu'elles ont lieu peu après la présentation d'une nouvelle cible dans l'environnement.

2.7 Problème de navigation d'un robot mobile

2.7.1 Introduction

Dans cette section, nous donnons un exemple applicatif de l'algorithme CbL, dans le cadre d'un problème de viabilité. Le problème de navigation d'un robot mobile consiste à atteindre un objectif, en ayant un comportement de suivi de mur lorsqu'on ne peut pas se rendre directement vers l'objectif. C'est ce comportement d'évitement qui est appris et qui forme un problème de viabilité. Puis, il est utilisé au sein d'un algorithme de haut niveau.

Nous montrons que plusieurs agents, appelés CbMU pour Constraint based Memory Units, s'adaptant grâce à l'algorithme CbL, peuvent se connecter hiérarchiquement de manière à induire le comportement d'évitement. L'objectif est, dans ce cas, de montrer qu'on peut réduire l'espace d'états pour chacun de ces agents, en les spécialisant dans des tâches perceptives particulières.

2.7.2 Position du problème

Le robot mobile miniature Khepera (figure 2.11) est circulaire et mesure environ 5 cm de diamètre. Il possède 8 capteurs infrarouge $s_1, \ldots, s_8$, dont la portée maximum est de 5 cm[1] et dont les données sont significativement bruitées. Ces capteurs renvoient des données codées sur 10 bits (de 0, à 1023) : « 0 » signifie que le capteur ne signale pas d'obstacle, alors que « 1023 » signale un obstacle très proche. Il faut préciser à ce sujet que la portée minimale du capteur est d'environ 2 cm. Le robot est commandé en envoyant deux directives concernant les vitesses linéaires ls_1 et ls_2 de ses roues. Vue la légèreté de ce robot et la faible vitesse de déplacement, l'inertie de ce dernier est négligeable. Pour de plus amples informations sur ce robot, le lecteur pourra consulter [Mondada et al., 1994].

Pour notre expérience, nous avons utilisé l'environnement de simulation de robot mobile Khepera, programmé par Michel [Michel, 1996]. Le simulateur reproduit correctement les imperfections des mesures issues des capteurs, rendant les résultats obtenus par le simulateur très proches de ceux du robot réel (voir [Maaref et al., 1999] pour des résultats qualitatifs).

Le problème de navigation du robot simulé consiste à atteindre un objectif en utilisant un comportement de suivi de mur (problème de viabilité) lorsqu'un obstacle l'empêche d'aller directement vers celui-ci. On suppose que les positions du robot et de l'objectif sont parfaitement connues à tout moment. On suppose qu'il existe deux signaux de renforcement permettant de

1. Cette valeur dépend des conditions de l'expérience. Nous considérons ici des obstacles blancs, avec un éclairage d'intérieur, c'est-à-dire environ 300 lux.

résoudre le problème d'évitement, nommés r_1 et r_2, dont les fonctions respectives sont de savoir si le robot s'est cogné contre un obstacle et si le robot est resté trop longtemps éloigné d'un obstacle.

La sous-section suivante présente une démarche de création de contexte pour que le problème d'évitement puisse être résolu par l'algorithme CbL. Comme il s'agit d'un problème de viabilité, nos résultats théoriques indiquent qu'il suffit que le problème de décision soit markovien ; Nous noterons en particulier que le contexte ne respecte pas (P_ϵ).

2.7.3 Préparation du contexte d'apprentissage pour le problème de navigation du robot Khepera

L'ensemble des graphes de cette sous-section est donné par la figure 2.12.

Dans le cadre d'un problème de viabilité, l'utilisation de l'algorithme CbL nécessite que le problème de décision soit markovien, c'est-à-dire que la probabilité de transition d'un état $e_{i,k}$ vers un état e_j ne dépende que de e_i et de a_k. L'échec de l'algorithme signifie soit qu'il n'existe pas de politique de commande fiable (à partir des états que nous avons constitués *a priori*), soit que le problème de décision n'est pas markovien.

Nous allons tout d'abord donner un premier contexte d'apprentissage, nommé C_1. Nous montrerons qu'il ne respecte pas la propriété (P_ϵ). Des résultats expérimentaux préliminaires (que nous ne donnons pas dans ce document) montrent que l'algorithme CbL échoue avec ce contexte. Nous présentons alors un deuxième contexte d'apprentissage, nommé C_2, utilisant une méthode de découpage hiérarchique de la tâche d'évitement. Nous montrons que l'algorithme CbL est alors capable de déterminer une politique de commande fiable, même si C_2 ne respecte pas la propriété (P_ϵ).

Dans cette sous-section, nous expliquons comment nous construisons les contextes C_1 et C_2 et nous prouvons que ceux-ci ne respectent pas la propriété (P_ϵ). Notre démarche est constituée des étapes suivantes :

1. déterminer la finesse du découpage sur chaque axe de l'espace d'entrée du système, qui est *a priori* de dimension 8

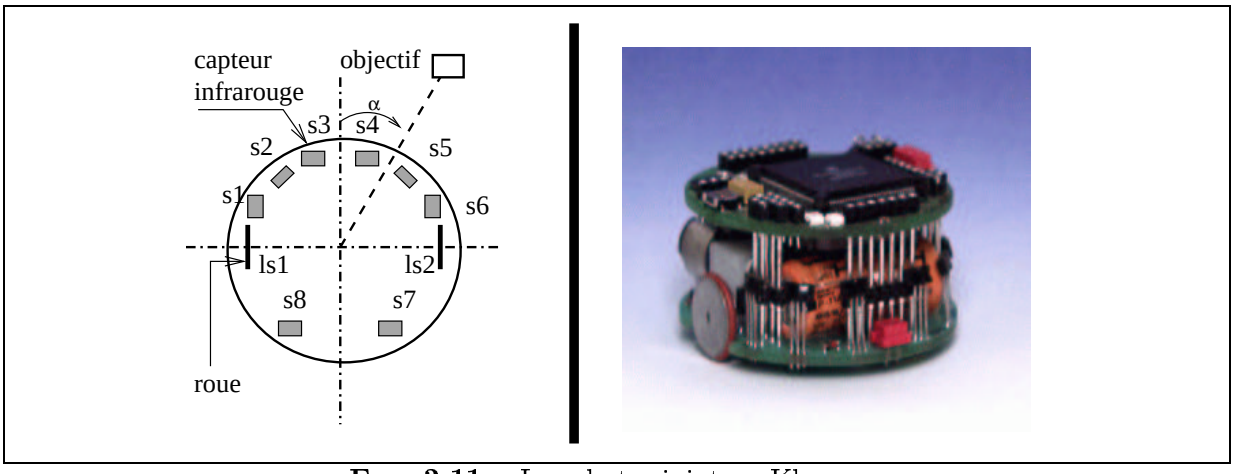


Fig. 2.11 – Le robot miniature Khepera.

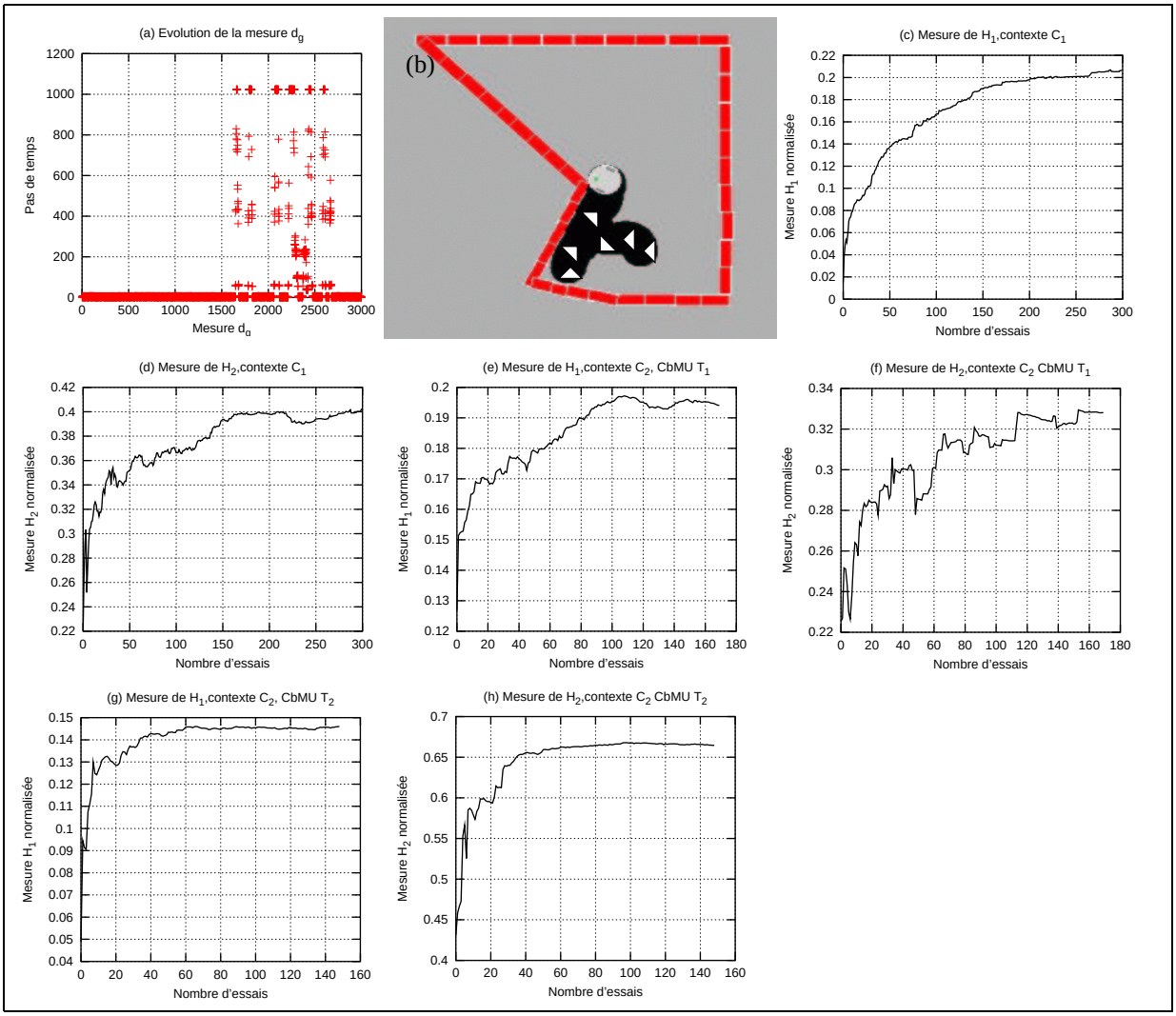


Fig. 2.12 – Graphes de la sous-section 2.7.3.

TAB. 2.1 – Commandes de base du robot Khepera simulé. Les valeurs de ls_1 et ls_2 sont sans unité.

Comportement	Signification	valeur de ls_1	valeur de ls_2
a_1	Tout droit	3	3
a_2	Avancer à droite	2	0
a_3	Avancer à gauche	0	2
a_4	Tourner sur la droite	2	-2
a_5	Tourner sur la gauche	-2	2

2. éliminer si possible les redondances entre les signaux d'entrée, de manière à réduire la dimensionnalité du problème
3. déterminer l'ensemble des commandes de base du robot

Imaginons que nous souhaitions partitionner l'espace d'entrée du système (qui est de dimension 8, puisque le robot est doté de 8 capteurs) en un ensemble de boîtes, tout comme dans l'exemple du pendule inversé (voir le chapitre 1). Le problème est de savoir comment il faut découper selon chacune des dimensions. Faut-il prendre un découpage fin où pas? Le graphe (a) montre l'évolution du capteur s_1 dans le temps (graphe (a)), lorsque le robot évolue dans l'environnement donné par le graphe (b). Nous constatons des sauts de valeur très brusques sur s_1. En conséquence, nous supposons qu'un découpage très grossier selon chaque axe de l'espace d'entrée est suffisant. Cela a déjà été constaté par [Maaref et al., 1999] entre autres, dans le cadre de la navigation d'un robot mobile Khepera utilisant un système d'inférence flou. Nous découperons en 4 suivant chaque axe. Ainsi, le découpage sera le suivant :

$$[0,1023] = [0,255] \bigcup [256,511] \bigcup [512,767] \bigcup [768,1023]$$

Malgré ce faible découpage, le nombre d'états est de $8^4 = 4096$. Nous souhaitons réduire ce nombre. Pour cela, nous constatons que les valeurs associées à des capteurs voisins sont fortement corrélées : s_1 et s_2, s_3 et s_4, s_5 et s_6, puis s_7 et s_8. Nous allons réduire la dimensionnalité du problème en choisissant, pour chacune des paires de capteurs, le maximum des deux valeurs (c'est-à-dire le capteur donnant la plus courte distance). Les valeurs effectivement utilisées pour déterminer les états du système seront les suivantes :

$$d_g = \max\{s_1,s_2\} \ , \ d_h = \max\{s_3,s_4\} \ , \ d_d = \max\{s_5,s_6\} \ , \ d_b = \max\{s_7,s_8\}$$

Le nombre d'états est alors ramené à $4^4 = 256$. Nous choisissons à présent l'ensemble des commandes de base du robot, c'est-à-dire un ensemble de couples (ls_1,ls_2). Nous en avons sélectionné 5, nommées a_1,$a2$,...,$a5$ (voir le tableau 2.1 pour un descriptif), qui nous semblent suffisantes *a priori* pour accomplir la tâche d'évitement. L'ensemble des 256 états et des 5 commandes de base du robot constitue le contexte d'apprentissage C_1 de l'algorithme CbL.

Le calcul des mesures H_1 et H_2 utilise l'algorithme 1.1, page 17. L'objectif est de découvrir l'ensemble des transitions possibles entres les états $e_{i,k}$ et les états e_j du système. Ces transitions représentent les changements d'états possibles, suivant l'exécution des 5 commandes de base. Pour que H_1 et H_2 soient représentatives de l'ensemble de ces changements, il est nécessaire de placer le robot dans des environnements différents : nous en avons utilisé 10. Pour chacun de ceux-ci, nous effectuons 30 essais (voir l'algorithme 2.3). À la fin de chaque essai, on calcule les valeurs de H_1 et H_2. L'ensemble de ces valeurs forme les graphes (c) (pour H_1) et (d) (pour H_2). On constate que la propriété (P_ϵ) (voir la sous-section 1.3.4, page 12) n'est pas satisfaite,

Algorithme 2.3 Description d'un essai permettant le calcul de H_1 et de H_2

```
Initialisation au hasard du robot dans l'espace libre de son environnement
Répéter N fois
    Récupérer l'état e_i du système en fonction des données capteur
    Répéter
        Choisir une commande a_k au hasard (voir le tableau 2.1)
        Exécuter a_k
        Récupérer l'état résultant e_j
    Jusqu'à e_j ≠ e_i
    Incrémenter le nombre de transitions entre e_{i,k} et e_j
    [Se reporter à l'algorithme 1.1, page 17]
FinRépéter
```

car les valeurs de H_1 et de H_2 ne sont pas proches de 0 : la courbe de H_1 semble posséder une limite supérieure proche de 0.2. La courbe de H_2 s'approche très rapidement de la valeur 0.4 . La courbe de H_1 est caractérisée par des « sauts » qui représentent l'expérience d'une nouvelle zone de l'espace d'états.

Le découpage hiérarchique a pour objet de transformer le contexte d'apprentissage du réflexe d'évitement en sous-contextes plus facilement apprenables. Cela revient à considérer un ensemble de sous-graphes construits à partir du graphe d'états induit par C_1, de manière à diminuer le nombre de transitions partant d'un état $e_{i,k}$. Cela suppose que la nature de la transition entre $e_{i,k}$ et e_j n'est pas objectivement aléatoire, mais qu'elle dépend essentiellement d'un contexte. Le graphe (i) donne un exemple pour lequel le problème de transitions multiples peut être résolu en considérant deux contextes différents.

Le découpage de la tâche globale (atteinte d'une objectif en évitant les obstacles) s'organise tout d'abord autour de deux grandes catégories de processus :

1. les processus pouvant être programmés « à la main », sans avoir à utiliser une méthode d'apprentissage
2. les processus devant être appris

Dans notre cas, nous supposons que la position de l'objectif, la position du robot et son orientation par rapport à un repère fixe sont connus parfaitement à tout moment : le processus consistant à diriger le robot vers l'objectif peut donc être programmé très facilement. Il reste le processus d'évitement par suivi de mur : il sera appris. Celui-ci est découpé en sous-tâches, qui seront apprises par des agents appelés CbMU (pour Constraint [1] based Memory Unit). Ces agents sont spécialisés de manière à réduire autant que possible leur nombre d'états. Les tâches à apprendre (décrites dans le tableau 2.2) sont hiérarchiquement connectées, suivant la figure 2.13. Chaque tâche est considérée comme un module indépendant, activé par une tâche hiérarchiquement supérieure. Lorsqu'un agent est activé, il choisit une des actions disponibles suivant la valeur courante des signaux perceptifs, puis reçoit en réponse un signal de renforcement qui lui est spécifique. Chaque tâche est réalisée par un agent CbMU, utilisant l'algorithme d'apprentissage CbL. L'ensemble des modules participe donc au comportement global d'évitement d'obstacle qui, à chaque pas de temps, revient à exécuter l'une des cinq actions élémentaires. Ce comportement

1. Le terme *Constraint* est dû à l'application de l'algorithme CbL sur chacun des agents.

Algorithme 2.4 Algorithme de haut niveau régissant la navigation du robot Khepera

[Contexte d'atteinte d'objectif]
Fonction AtteinteObjectif()
Si l'objectif est devant le robot et que celui-ci peut aller devant, alors
a_1 est exécuté
Sinon Si le robot est près d'un obstacle sur la gauche et que l'objectif est dans la direction de celui-ci, alors
Appel à la fonction SuiviMurGauche() (changement de contexte)
Sinon
Appel à la fonction SuiviMurDroite()
FinSi

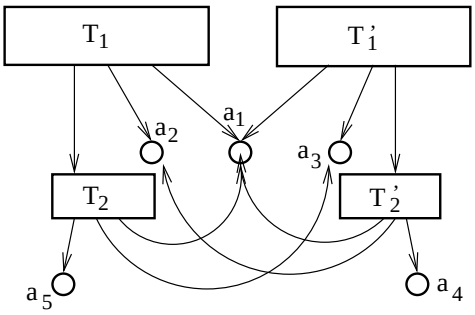
[Contexte de suivi de mur sur la gauche]
Fonction SuiviMurGauche()
Si l'objectif est sur la gauche du robot et qu'il peut aller à gauche sans se cogner OU
Si l'objectif est devant le robot et celui-ci peut aller devant sans se cogner, alors
Appel de la fonction AtteinteObjectif()
Sinon
L'action a_k est celle délivrée par l'agent associé au comportement « suivre un mur sur la gauche »
Exécuter a_k
FinSi

[Contexte de suivi de mur sur la droite]
Fonction SuiviMurDroite()
Si l'objectif est sur la droite du robot et qu'il peut aller à droite sans se cogner OU
Si l'objectif est devant le robot et celui-ci peut aller devant sans se cogner, alors
Appel de la fonction AtteinteObjectif()
Sinon
L'action a_k est celle délivrée par l'agent associé au comportement « suivre un mur sur la droite »
Exécuter a_k
FinSi

TAB. 2.2 – Description des agents CbMU participant à la tâche globale d'évitement

CbMU	Description de la tâche	signaux d'entrée	commandes possibles	contrainte associée
T_1	Suivi mur gauche	d_g,d_h,d_d	T_2,a_1,a_2	NPSC,MG
$T_1^{'}$	Suivi mur droite	d_g,d_h,d_d	$T_2^{'}$,a_1,a_3	NPSC,MD
T_2	Suivi mur convexe gauche	d_g	a_1,a_3,a_5	NPSC,MG
$T_2^{'}$	Suivi mur convexe droite	d_d	a_1,a_2,a_5	NPSC,MD

NPSC : « Ne pas se cogner », « MG » : être assez près d'un mur sur sa gauche, « MD » : être assez près d'un mur sur sa droite.



Les flèches signifient « peut utiliser ». Ainsi, la tâche T_1 (suivi de mur sur la droite) est une séquence de trois tâches plus élémentaires : a_1,a_2 et T_2. a_1 et a_2 sont non décomposables. Par contre, T_2 (évitement d'obstacle sur la gauche) est une séquence des trois actions élémentaires a_1,a_3 et a_5.

FIG. 2.13 – Hiérarchisation des agents utilisés par Khepera dans son comportement d'évitement d'obstacles

global est utilisé par un algorithme de haut niveau d'atteinte d'objectif (algorithme 2.4), composé d'une base de règles, dont les prémisses utilisent les valeurs des marquages (obtenues grâce à l'apprentissage) des différents modules. L'objectif de ces règles est de constituer un branchement permettant de décider, à tout moment, si on peut se diriger vers l'objectif en toute sécurité, ou s'il faut adopter un comportement de suivi de mur. Il faut noter que la création d'une transition dans un graphe associé à une CbMU n'est possible que si celle-ci est activée (appelée par un supérieur hiérarchique ou par l'algorithme de haut niveau 2.4).

Nous venons donc de fixer C_2, qui est composé d'un ensemble de sous-contextes dans lesquels évoluent quatre agents CbMU. Nous allons à présent calculer les valeurs de H_1 et de H_2 pour les CbMU T_1 et T_2. Les deux autres étant leur « symétrique », elles sont *a priori* associées aux mêmes valeurs de H_1 et de H_2. Pour cela, nous effectuons la même démarche que précédemment, en choisissant au hasard : 1) la tâche d'évitement à accomplir, parmi T_1 et $T_1^{'}$, 2) en choisissant l'action à exécuter au hasard ; si celle-ci consiste à activer une autre CbMU (tâches T_2 ou $T_2^{'}$), on choisit pour cette dernière l'action à exécuter au hasard. Les résultats sont présentés dans les graphes (e) et (f). On remarque que le découpage de la tâche principale (évitement en suivant un mur à droite ou à gauche) ne permet pas de réduire les valeurs de H_1 et de H_2. Par contre, on constate que la vitesse de « convergence » de H_1 et de H_2 vers leur valeur maximale est sensiblement plus rapide grâce au découpage. Cela signifie que l'ensemble des transitions possibles pour chacun des contextes des CbMU est plus rapidement trouvé. Or, on sait que la vitesse de convergence de l'algorithme CbL dépend uniquement de la rapidité avec laquelle le système ajoute des transitions dans le graphe perceptif. Donc, avant d'avoir effectué l'apprentissage, on peut prédire que la convergence de l'apprentissage sera plus rapide dans le contexte C_2 que dans

C_1.

2.7.4 Protocole expérimental

Notre objectif est d'étudier l'apprentissage du comportement d'évitement par suivi de mur. Le problème d'atteinte d'objectif est ensuite réglé en utilisant l'algorithme de navigation 2.4, une fois que l'apprentissage est terminé.

L'algorithme CbL(1) sera utilisé pour ce problème de viabilité. L'ensemble des marquages possibles est réduit à {0,-1}.

Nous effectuons 100 apprentissages. Chacun de ceux-ci comporte 1000 essais. Chaque essai comporte 500.000 itérations. Un essai est considéré comme réussi si le robot a respecté ses contraintes de viabilité pendant 500.000 itérations consécutives.

On effectue en premier l'apprentissage des agents de plus bas niveau, c'est-à-dire T_2 et T_2'. Pour cela, au début de chaque essai, on initialise la position du robot d'une manière aléatoire dans l'espace libre de l'environnement. Les deux apprentissages sont effectués indépendamment. Dans un deuxième temps, on effectue l'apprentissage de T_1 et de T_1' en utilisant les agents T_2 et T_2' ayant appris. Pour ces deux apprentissages, on initialise le robot près d'un mur, de manière à ce que les conditions initiales soient favorables (restent dans la zone de viabilité du problème de suivi de mur).

2.7.5 Résultats

Nous allons d'abord donner les résultats d'apprentissage de l'algorithme CbL. Le comparatif avec la méthode du Q-Learning est de même nature que pour le problème précédent. Nous le mentionnerons en fin d'analyse.

L'apprentissage est effectué dans l'environnement donné par la figure 2.14. Nous montrerons par la suite les capacités du robot à se mouvoir dans un autre environnement.

Intéressons-nous tout d'abord à l'apprentissage de la tâche T_1. Les 100 apprentissages grâce à l'algorithme CbL se sont tous terminés par un succès, obtenu après un nombre d'essais variant entre 35 et 42. La figure 2.15 résume les résultats pour un de ces 100 apprentissages. Le graphe (a) montre l'évolution de la durée de viabilité suivant le nombre d'essais effectués. Il apparaît que la durée de viabilité augmente essentiellement aux moments où on explore de nouveaux états du systèmes (graphe (c)). Dans ce cas, le nombre maximal de sommets du graphe est $64 + 3 \times 64 + 1 = 257$[1]. On s'aperçoit que ce nombre est quasiment atteint à la fin de l'apprentissage, ce qui signifie que l'exploration de l'espace de perception a été presque totale (graphe (b)). Enfin, l'apprentissage de T_2 a demandé entre 25 et 33 essais.

À titre de comparaison, les résultats obtenus avec la méthode du Q-Learning mettent en évidence un temps d'apprentissage environ 7 fois plus important que pour CbL : il faut en moyenne

1. D'après le tableau 2.2, page 75, T_1 utilise trois variables d'entrée (d_g,d_h,d_d), qui sont chacune divisées en 4 intervalles : le nombre d'états e_i est donc $4^3 = 64$. D'autre part, trois commandes sont possibles, ce qui donne un nombre d'états $e_{i,k}$ égal à 3×64. Le dernier état est l'état terminal. D'où un nombre total d'états du graphe de 257.

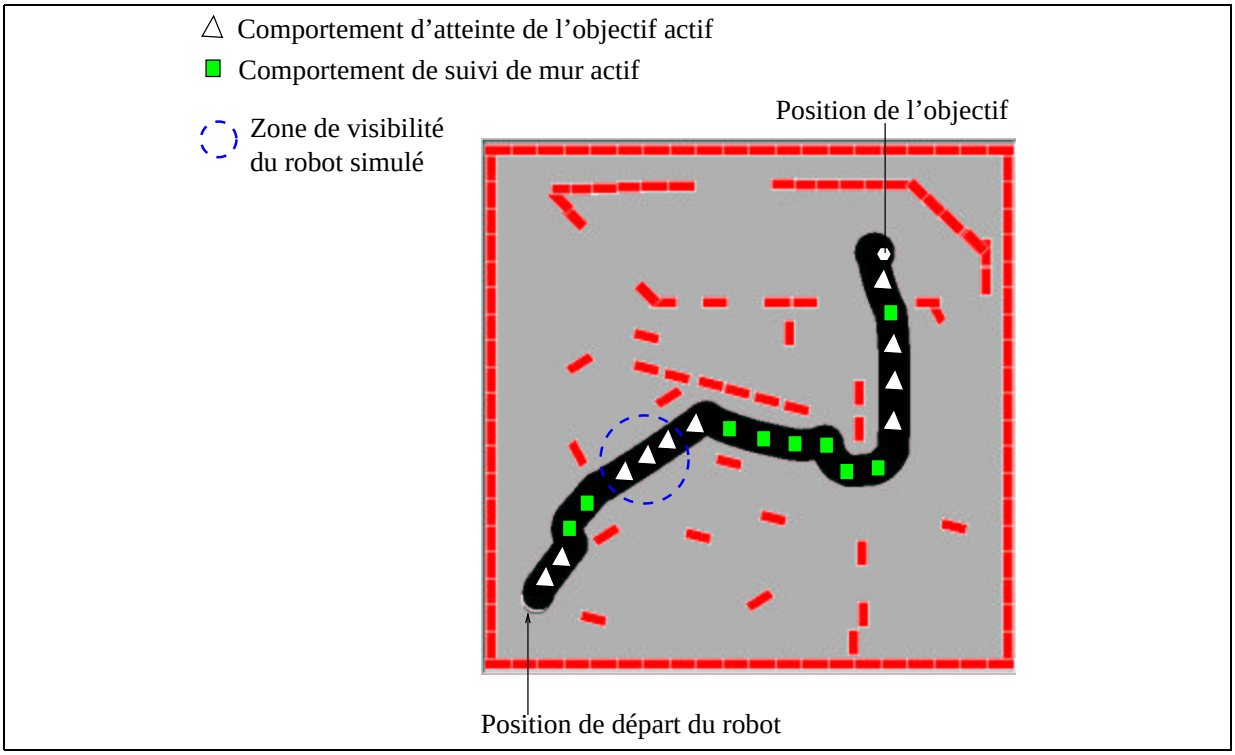


FIG. 2.14 – Comportement de recherche d'objectif avec évitement d'obstacles de Khepera.

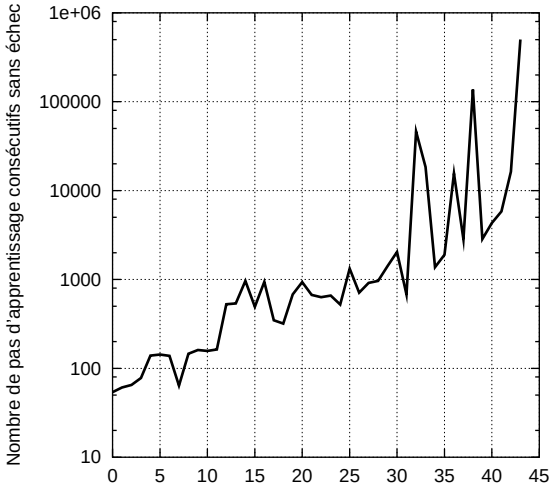
182 essais pour achever T_1 et 304 essais pour achever T_2.

Lorsque les quatre agents ont subi un apprentissage, on les utilise au sein de l'algorithme d'atteinte d'objectif 2.4 (voir la page 74). Le résultat visuel du comportement du robot est donné par la figure 2.14.

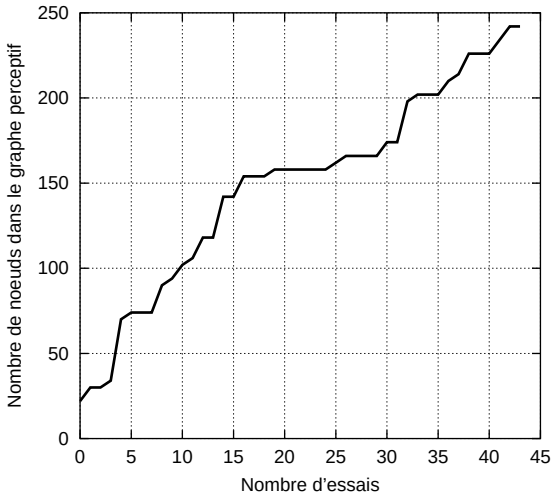
Comment évolue le comportement du robot au cours de l'apprentissage ? Nous avons constaté que le robot commence par suivre les murs au plus près. Or, cette stratégie ne garantit pas une sécurité totale dans tous les cas : en effet, les capteurs du robot peuvent ne pas rendre compte d'une petite aspérité au niveau d'un mur (figure 2.16), car il existe des zones « aveugles ». Ces expériences « malheureuses » vont conduire le robot à suivre le mur d'un peu plus loin, en créant autour de lui une zone de sécurité assez importante pour éviter ces situations pour lesquelles la qualité de sa perception ne lui permet pas de garantir d'une manière certaine un parcours sans collision. Bien entendu, cet éloignement est régulé par la contrainte de ne pas trop s'éloigner d'un mur. Donc, le robot doit trouver, si cela est possible, un compromis entre ces deux contraintes et c'est ce qu'il arrive à faire.

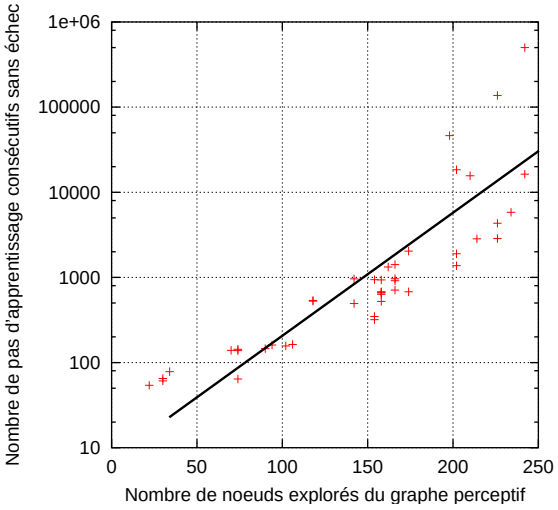
Le comportement du robot à un moment donné est donc en cohérence avec l'ensemble de ses expériences passées : l'ajout d'un nouveau cas particulier peut provoquer un changement de comportement très net (par la découverte d'une transition dans le graphe d'état d'une CbMU). La fréquence d'apparition de ces cas n'a aucune influence sur le résultat final, puisque la contrainte d'équilibre ne pousse le système à modifier les marquages des états qu'au premier cas rencontré : cela se traduit par l'ajout d'une connexion dans le graphe perceptif, qui peut engendrer une rupture de la cohérence induite par cette contrainte.

(a) Evolution de la durée de viabilité pour un apprentissage



(b) Evolution du nombre d'états explorés





(c) Relation entre le nombre d'états explorés et la durée de viabilité

Commentaires :
Le graphe (a) montre l'évolution du nombre d'itérations viables suivant les essais, pour un apprentissage réussi, qui se termine au bout de 42 essais (500.000 itérations consécutives viables). Le graphe (b) présente, pour cet apprentissage, l'évolution du nombre d'états parcourus par le système au cours des essais. Enfin ,le graphe (c) met en évidence la relation qui existe entre la découverte de nouveaux états et l'augmentation de la durée de viabilité. La droite correspond à la meilleure modélisation de cette relation, suivant le critère des moindres carrées, par une fonction de la forme exp(ax+b), avec x le nombre d'états explorés du graphe, « a » étant la pente de la droite et « b » sa position à l'origine.

FIG. 2.15 – Résultats d'apprentissage concernant l'agent T_1.

FIG. 2.16 – Cas d'échec amenant le robot à s'éloigner du mur.

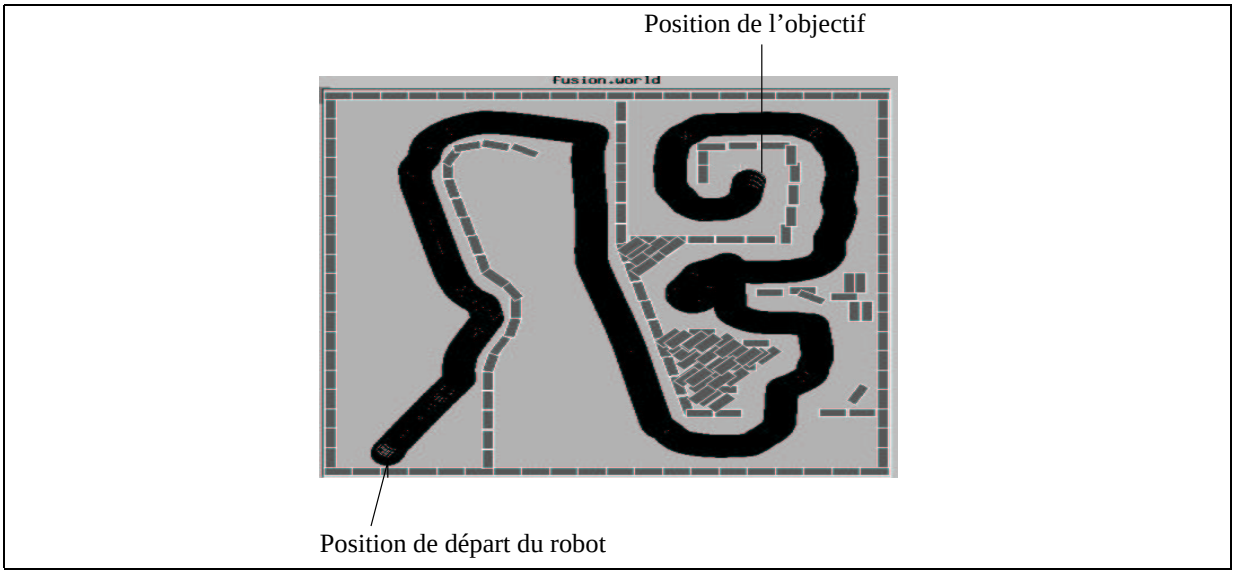


FIG. 2.17 – Nouvel environnement présenté au robot.

Le changement d'environnement ne pose pas de problème (figure 2.17). Dans chacun des cas, le robot Khepera finit par respecter l'intégralité de ses contraintes à chaque pas de temps. En effet, l'idée sous-tendant l'algorithme est que l'apprentissage consiste à intégrer une série de cas particuliers dans un ensemble gardant toujours sa cohérence en fonction de contraintes particulières. Le traitement d'un nouveau cas est effectué dès son apparition (ajout d'une nouvelle connexion dans le graphe perceptif) et peut produire un changement du marquage de certains états du graphe perceptif. Lorsque ce cas se représentera, aucune modification des marquages ne se reproduira plus (pas d'ajout de connexion, celle-ci existant déjà).

Dans ce contexte, lorsque le robot est placé dans un nouvel environnement, un apprentissage a lieu si cet environnement comporte des cas perceptifs jamais rencontrés auparavant (exploration de nouveaux états ou découverte de nouvelles connexions entre des états déjà explorés). Il est important de noter que l'ajout d'un nouveau cas ne provoque pas l'oubli des cas déjà rencontrés : ainsi, si on replace le robot dans un ancien environnement, le robot sera toujours capable de remplir ses contraintes, mais il le fera peut-être d'une manière plus « prudente ».

On peut utiliser plusieurs environnements pour faire face plus rapidement à l'ensemble des situations perceptives. L'idée est alors de « provoquer » la découverte rapide des transitions du graphe d'états, ce qui permet d'accélérer la convergence de l'algorithme CbL. Pour cela, nous construisons un ensemble d'environnements possédant chacun une particularité perceptive (angle aigu, couloir étroit, etc.). Nous connaissons les capacités d'incrémentalité de l'algorithme CbL et nous pouvons les utiliser pour construire un ensemble de « leçons » à l'élève Khepera. Par exemple :

1. suivi d'un mur continu convexe (figure 2.18 (a))
2. suivi d'un mur continu quelconque (figure 2.18 (b))
3. suivi d'un mur dans un environnement possédant des portes (figure 2.18 (c))
4. suivi d'un mur dans un environnement possédant des couloirs (figure 2.18 (d))
5. suivi d'un mur possédant de petites aspérités (figure 2.18 (e))
6. suivi d'un mur possédant de grandes aspérités, pouvant ressembler à des portes ou à des

couloirs (figure 2.18 (f))

Cette technique permet de se « concentrer » sur un type de perception donné afin d'exercer spécifiquement le robot. Dans notre cas de figure, cela n'a pas permis de réduire le nombre d'essais nécessaires à la réussite de l'apprentissage, car celui-ci est déjà très bas. Par contre, dans des cas plus complexes, cette stratégie permet de placer rapidement le robot dans une situation particulière, même si celle-ci est peu fréquente dans un environnement « moyen ». Si cette situation n'est pas maîtrisée, le robot échoue donc après un temps court. La durée de la leçon va représenter le temps qu'il faut pour résoudre la difficulté spécifique.

Enfin, il nous semble important d'effectuer une remarque concernant l'algorithme de haut niveau 2.4. Cette remarque porte sur **l'utilisation d'une séquence de commandes apprises pour discriminer des catégories perceptives**[1] (c'est précisément l'un des enjeux de l'AP). L'algorithme de haut niveau spécifie le choix parmi trois contextes différents (aller à l'objectif, suivre le mur sur la droite, suivre le mur sur la gauche) en fonction du positionnement du robot par rapport à son environnement : « si le robot peut aller sur sa gauche » signifie « le robot a-t-il la place de se déplacer sur sa gauche sans se cogner ». Or, la réponse à cette question est donnée par l'expérience de la contrainte « ne pas se cogner », qui est apprise. La décision entre les trois contextes peut donc s'effectuer finement si le robot a acquis la capacité de se déplacer (sans objectif précis) sans se cogner dans son environnement. Nous voyons donc ici une illustration de l'apprentissage à deux niveaux, pour lequel l'apprentissage de bas niveau (navigation sans chocs) sert à un choix de plus haut niveau (atteindre un objectif précis). Nous avons signalé que le robot commençait par suivre un mur de très près, puis s'en éloignait après des expériences « malheureuses ». En conséquence, sa notion de « Je peux aller sur ma gauche sans choc » évolue également, puisque son espace de sécurité augmente. Donc la prise de décision d'un contexte va être modifiée, non pas par l'apprentissage de cette décision en elle-même, mais par un apprentissage de bas niveau. Celui-ci correspond à un **apprentissage perceptif élémentaire**.

2.8 Conclusion

Nous avons construit un algorithme d'AO, appelé CbL pour Constraint based Learning, qui est fondé sur la réaction du système (qui est un graphe d'états possédant chacun un marquage) à un ensemble de contraintes. Nous avons achevé l'ensemble des étapes de notre méthodologie :

- choix d'une contrainte qui s'applique au système d'AO, qui est dérivée de l'algorithme *Minimax*
- découverte de la manière d'assurer le respect de cette contrainte à tout moment
- preuve que l'interaction avec l'environnement peut s'interpréter comme un apprentissage
- preuve que cet apprentissage est prédictible et que, s'il existe une solution à celui-ci, elle est fiable

Les caractéristiques de prédictibilité et de fiabilité sont donc émergentes à partir de la spécification de notre contrainte d'équilibre. L'adaptation du système à une modification de l'environnement est également une propriété de l'algorithme CbL. Cependant, l'ensemble de ces résultats n'est valable que si le contexte de l'AO est idéal (c'est-à-dire s'il vérifie (P_ϵ)). Mais, l'influence du contexte est ici limitée à la topologie des états du système. En effet, l'algorithme CbL ne comporte pas de paramètre susceptible de modifier le résultat de l'apprentissage.

1. Comprendre le rôle de l'action sur la genèse de la perception est un thème actuel important en sciences cognitives.

Des exemples applicatifs simples appuient nos résultats théoriques. Nous montrons que **l'algorithme CbL converge nettement plus rapidement que l'adaptation $Q(\lambda)$ de l'algorithme du Q-Learning** (10 fois pour un problème simple et beaucoup plus pour un problème plus complexe). Ces résultats confirment **l'intérêt d'utiliser un apprentissage latent** qui, comme l'a déjà été noté par Stolzmann, peut être utilisé avec avantage en combinaison de techniques de renforcement classiques. Dans notre cas, il serait instructif de comparer nos résultats avec ceux obtenus après utilisation de l'*ACS* de Stolzmann.

D'autre part, **nous montrons les capacités d'incrémentalité de CbL**. Lorsque la phase d'exploration est complète (toutes les transitions du graphe d'état ont été explorées au moins une fois), **la politique de commande est optimale**, sauf dans des cas très particuliers qui peuvent être surmontés.

La première partie de notre travail est donc achevée avec succès. Il nous reste à aborder la partie la plus délicate, concernant l'AP. Nous rappelons que l'objectif de celle-ci est de permettre la création d'un contexte idéal (au sens de la contrainte (P_ϵ) que nous avons définie dans le premier chapitre.

Références

Anderson, C. (1989). Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine*, 9(3) :31–37.

Barto, A., Sutton, R., and Anderson, C. (1983). Neurolike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC13 :834–846.

Davesne, F. and Barret, C. (1999a). Constraint based memory units for reactive navigation learning. In *European Workshop on Learning Robots*.

Davesne, F. and Barret, C. (1999b). Reactive navigation of a mobile robot using a hierarchical set of learning agents. In *IROS'99*, Kyongyu, Corée.

Glorennec, P. (2000). Reinforcement learning : an overview. In *European Symposium on Intelligent Techniques (ESIT'2000)*.

Maaref, H., Barret, C., and Amamou, A. (1999). Optimization of a fuzzy controller for a reactive navigation. *Computational Intelligence and Applications*, pages 193–197.

Michel, O. (1996). Khepera simulator package version 2.0 : Freeware mobile robot simulator. `http://wwwi3s.unice.fr/~om/khep-sim.html`.

Mondada, F., Franzi, E., and Ienne, P. (1994). Mobile robot miniaturization : A tool for investigation in control algorithms. In Yoshikawa, T. and Miyazaki, F., editors, *Proceedings of the Third International Symposium on Experimental Robotics 1993*, pages 501–513. Springer Verlag,.

Peng, J. and Williams, R. (1996). Incremental multi-step q-learning. *Machine Learning*, 22 :283–290.

Rich, E. (1983). *Artificial Intelligence*. McGraw-Hill.

Stolzmann, W. (1998). Anticipatory classifier systems. pages 658–664.

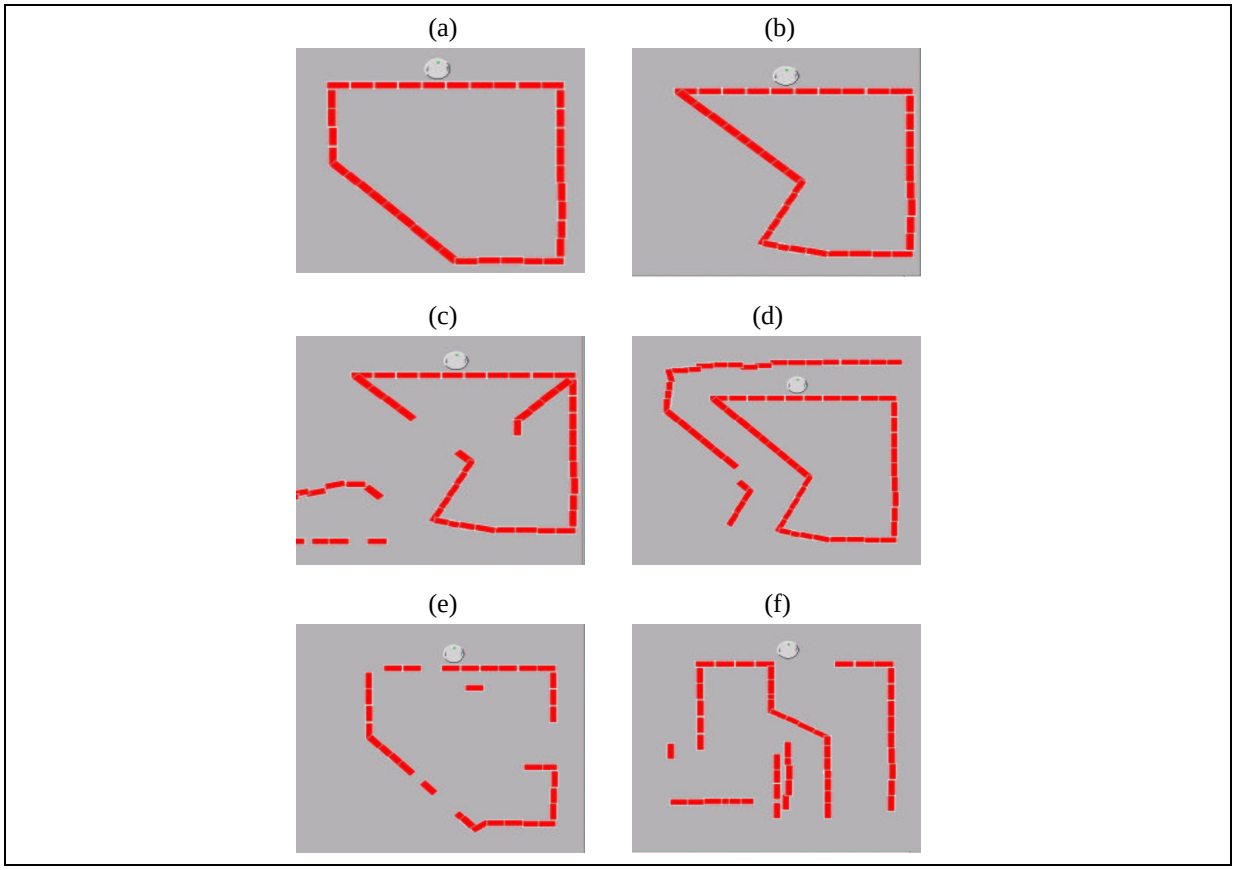


FIG. 2.18 – Exercices de difficulté progressive proposés au robot.

A

Éléments relatifs au chapitre 1

A.1 Techniques d'apprentissage par renforcement

A.1.1 Avertissement - remerciements

Les schémas et algorithmes présentés dans cette section sont extrait du recueil « Apprentissage par Renforcement », écrit par M. Glorennec (on pourra se référer à [Glorennec, 2000]). Nous tenons à le remercier pour l'aide que ce document nous a apportée.

A.1.2 Architecture et algorithme Q-Learning

L'algorithme Q-Learning est un exemple d'algorithme utilisant l'itération sur les valeurs qualité associées à chaque état (figure A.2). L'algorithme A.5 est celui de $Q(\lambda)$. Les paramètres internes sont γ, λ et β. Un paramètre est également introduit dans le mécanisme de choix d'action (le facteur de température T, s'il s'agit d'une méthode « à la Boltzmann »). e_t est un vecteur représentant la trace d'éligibilité. Celle-ci peut s'exprimer de deux manières différentes, selon les équations suivantes :

1. éligibilité accumulatrice :

$$e_t(x,a) = \begin{cases} 1 + \gamma\lambda e_{t-1}(x,a) & si\ x = x_t\ et\ a = a_t \\ \gamma\lambda e_{t-1}(x,a) & sinon \end{cases}$$

2. éligibilité remplaçante :

$$e_t(x,a) = \begin{cases} 1 & si\ x = x_t\ et\ a = a_t \\ \gamma\lambda e_{t-1}(x,a) & sinon \end{cases}$$

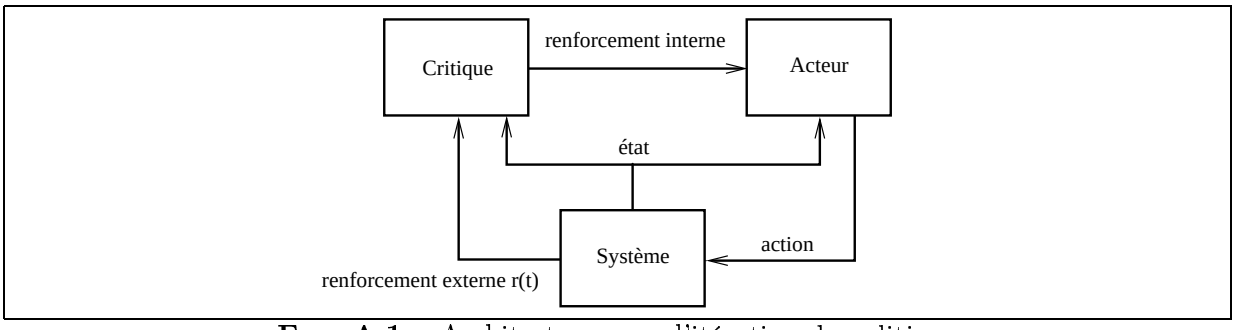


Fig. A.1 – Architecture pour l'itération de politique

Algorithme A.5 Algorithme du $Q(\lambda)$

t=0, Q(x,a)=0,$e_t(x,a) = 0$, pour tout état x et toutes action a
Observer x_t
répéter
choisir a_t
observer la transition de x_t vers x_{t+1}
$e_t^T := r_{t+1} + \gamma V_t(x_{t+1}) - Q(x_t,a_t)$
$e_t := r_{t+1} + \gamma V_t(x_{t+1}) - V_t(x_t,a_t)$
calculer e_t(x,a)
calculer $Q(x,a)$ pour tout état x et toute action a :
$Q(x_t,a_t) := Q(x_t,a_t) + \beta e_t^T e_t(x_t,a_t)$
$Q(x,a) := Q(x,a) + \beta e_t e_t(x,a)$
t := t+1
jusqu'à une condition d'arrêt

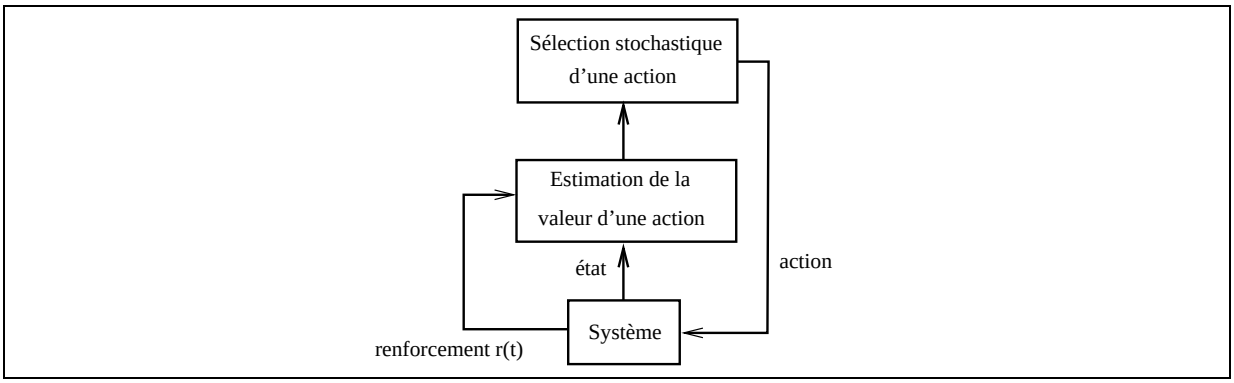


Fig. A.2 – Architecture pour l'itération sur les valeurs

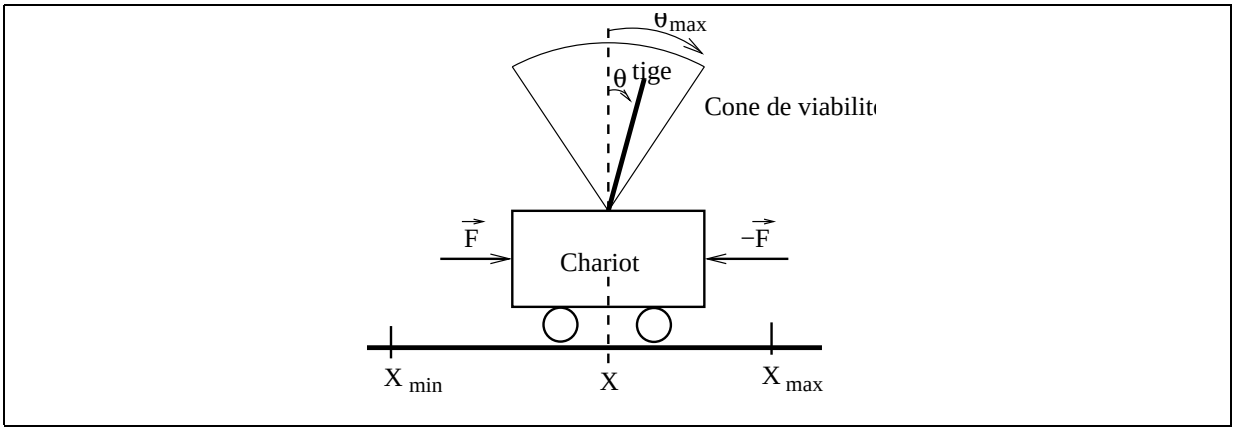


FIG. A.3 – Le problème du pendule inversé

A.2 Expériences autour du pendule inversé

A.2.1 Le problème du pendule inversé

La figure A.3 montre le système étudié. Il est composé d'un chariot possédant un degré de liberté (axe des X) et d'un pendule, en liaison rotule avec le chariot, dont on suppose qu'il possède également un degré de liberté, symbolisé par l'angle θ entre la tige et l'axe vertical du chariot.

Les équations de la dynamique du système sont reprises de [Anderson, 1989]. Les voici :

$$\ddot{\theta} = \frac{mg\sin\theta \; - \; \cos\theta\,(F + m_p l\dot{\theta}^2 \sin\theta)}{\frac{4}{3}ml - m_p l\cos^2\theta}$$

$$\ddot{x} = \frac{1}{m}\,(F + m_p l(\dot{\theta}^2 \sin\theta \;\; - \ddot{\theta}\cos\theta))$$

La simulation utilise la méthode d'Euler pour intégrer ces équations, avec un pas d'échantillonnage τ.

Nous reprenons le problème, tel qu'il a été décrit dans [Barto et al., 1983], avec les mêmes paramètres physiques (voir la table A.1). L'objectif est de maintenir à la fois le chariot entre les limites X_{min} et X_{max} et la position angulaire de la tige dans le cône de viabilité délimité par $-\theta_{max}$ et θ_{max}, en une des deux actions suivante à chaque pas de temps : pousser sur le chariot vers la gauche, avec une force F ou pousser sur la droite, avec une force F.

Dans [Barto et al., 1983], les données d'entrée sont les quatre variables d'état θ, x, $\dot{\theta}$, $\dot{x}$. L'espace d'états est découpé en boîtes suivant la table A.2, ce qui donne 162 boîtes.

A.2.2 Valeur des paramètres internes choisis pour l'algorithme Q-Learning

La valeur des paramètres dans l'algorithme Q(λ) est la suivante :

- $\gamma = 0.99$
- $\lambda = 0.8$
- $\beta = 0.5$

TAB. A.1 – Valeur des paramètres physiques du problème du pendule inversé

Masse du chariot m	1 kg
Masse de la tige m_p	0.1 kg
Demi-longueur de la tige l/2	0.5 m
Constante de pesanteur g	9.8 N.kg^{-1}
Pas d'échantillonnage τ	0.02 s
Valeur absolue F de la force appliquée au chariot	10 N
X_{max}	2.4 m
θ_{max}	12 deg

TAB. A.2 – Découpage de l'espace d'états

θ (deg)	<-6, [-6 ;-1[, [-1 ;0[, [0 ;1[, [1 ;6[, ≥ 6
$\dot{\theta}$ (deg.s^{-1}	<-50 , [-50 ;50[, ≥ 50
x (m)	<-0.8, [-0.8 ;0.8[, ≥ 0.8
$\dot{x}$ (m.s^{-1})	<-0.5, [-0.5 ;0.5[, ≥ 0.5

Pour le choix de l'action, nous utilisons une méthode « à la Boltzmann », dont le facteur de température T vaut 0.01 au début de l'apprentissage et diminue régulièrement, au début de chaque essai.

A.3 Éléments de calcul de probabilité

On considère un réel $\epsilon \in]0,1[$ ainsi que la variable aléatoire discrète X, définie par la loi suivante :

$$\forall k \in \mathbb{N}^* P(X = k) = \epsilon(1-\epsilon)^{k-1}$$

Et P(X=0) = 0.

Il s'agit bien d'une loi de probabilité sur $\mathbb{N}$ car $\sum_{k=0}^{\infty} P(X = k)$ existe et vaut 1.

L'espérance E[X] se calcule comme suit :

$$E[X] = \sum_{k=0}^{\infty} kP(X = k-1) = \sum_{k=0}^{\infty}(k+1)(1-\epsilon)^k$$

Or, la somme $\sum_{k=0}^{\infty}(k+1)r^k$, pour $r \in [0,1[$, est une somme classique et vaut $1/(1-r)^2$. D'où :

$$E[X] = 1/\epsilon$$

D'autre part, la variance Var[X] est donnée par l'expression suivante :

$$Var[X] = E[X^2] - (E[X])^2$$

Nous calculons $E[X^2]$:

$$E[X^2] = \sum_{k=0}^{\infty}(k+1)^2\epsilon(1-\epsilon)^k$$

Nous utilisons à nouveau le résultat de la somme classique suivante : $\sum_{k=0}^{\infty}(k+1)(k+2)r^k$ qui vaut $2/(1-r)^3$. En décomposant $E[X^2]$ pour faire apparaître cette somme, on obtient :

$$E[X^2] = (2-\epsilon)/\epsilon^2$$

On en déduit l'expression de Var[X] :

$$Var[X] = (1-\epsilon)/\epsilon^2$$

A.4 Calcul d'un estimateur $\hat{\epsilon}$ par la méthode du maximum de vraisemblance

On considère un réel $\epsilon \in]0,1[$ ainsi que la variable aléatoire discrète X, définie par la loi suivante :

$$\forall k \in \mathbb{N}^* P(X=k) = \epsilon(1-\epsilon)^{k-1}$$

Et P(X=0) = 0

On considère à présent un n-échantillon x_1,x_2,...,x_n issu de n observations de X. En supposant que ces observations sont indépendantes, la fonction $L(x_1,x_2,...,x_n|\epsilon)$ vaut :

$$L(x_1,x_2,...,x_n|\epsilon) = \prod_{k=1}^{n} P(X=x_k)$$

En prenant le logarithme népérien de L, pour simplifier les calculs, on obtient :

$$\ln L = \sum_{k=1} n(\ln\epsilon + x_k \ln(1-\epsilon))$$

Nous cherchons le maximum de L suivant la variable ϵ, lorsque x_1,x_2,...,x_n sont fixés. Il vient :

$$\frac{\partial \ln L}{\partial \epsilon} = \frac{n}{\epsilon} - \frac{\sum_{k=1}^{n} x_k}{1-\epsilon}$$

Pour $\epsilon \in]0,1[$, cette dérivée s'annule pour :

$$\epsilon = \frac{1}{\frac{\sum_{k=1}^{n} x_k}{n} + 1}$$

On vérifie que cette valeur correspond bien à un maximum. Pour cela, il faut regarder la dérivée seconde :

$$\frac{\partial^2 \ln L}{\partial \epsilon^2} = -\frac{n}{\epsilon^2} - \frac{\sum_{k=1}^{n} x_k}{(1-\epsilon)^2} < 0$$

On obtient donc un estimateur $\hat{\epsilon}$.

Deuxième partie

Apprentissage perceptif (AP)

3

Modèle paramétrique du sous-système d'AP

3.1 Introduction

3.1.1 Idées directrices

Notre modèle d'un système d'apprentissage d'actions réflexes est scindé en deux sous-systèmes : le premier s'occupe de l'apprentissage perceptif (AP), alors que le second traite de l'apprentissage de l'atteinte d'un objectif (AO). La première partie de ce document a concerné l'AO. Le résultat de l'AP est **la formation des états du système**.

Un schéma du modèle du sous-système d'apprentissage perceptif a été donné dans l'avant-propos (figure 1, page xxii). Il se fonde sur **une notion de l'information différente de celle de Shannon** : nous appelons la nouvelle notion **information perceptive**. Dans notre cas, l'information perceptive est le résultat d'un processus dynamique de sélection à partir d'un signal et d'un ensemble d'événements attendus concernant l'évolution future du signal. Cet ensemble est nommé **mémoire**.

Le sous-système étudié est la mémoire. La modification de celle-ci au cours du temps sera appelée **Apprentissage Perceptif**. Conformément à notre démarche, on suppose que le sous-système « mémoire » est soumis, à chaque pas de temps, à un ensemble de contraintes : celles-ci seront nommées **Contrainte d'Observabilité (CO)** et **Contrainte d'Unicité (CU)**. Le résultat de l'apprentissage perceptif est un changement de l'ensemble des informations perceptives détectables.

La mémoire est **un ensemble d'événements potentiellement détectables**, concernant **l'évolution future du signal** ; ceux-ci seront nommés **hypothèses d'évolution du signal** ou, plus simplement, **hypothèse**. La spécification d'une hypothèse sera donnée dans ce chapitre. L'information perceptive est un sous-ensemble inclus dans cet ensemble. Elle correspondra à l'état du système, qui est une donnée d'entrée du sous-système d'apprentissage d'objectif (chapitre 2).

L'information perceptive est déterminée par un mécanisme de sélection, que nous décrirons dans ce chapitre. Nous fournirons les algorithmes de sélection, dans le cas où l'ensemble d'hypo-

thèses est fini et dans un cas particulier pour lequel celui-ci est infini. Dans ce dernier cas, nous montrerons qu'il est possible, grâce à la théorie du calcul sur les intervalles, de déterminer de manière exacte et sûre deux ensembles encadrant l'information perceptive issue du mécanisme de sélection.

Les contraintes du sous-système sont fondées sur le postulat que **la détection d'une information perceptive est un événement rare en théorie et très fréquent, en pratique, après apprentissage**. Ce postulat est explicité dans ce chapitre, au travers du problème « D ».

3.1.2 Plan du chapitre

L'objectif de ce chapitre est de décrire un modèle paramétrique de cette mémoire. Le formalisme associé au respect des contraintes sera exposé dans le chapitre suivant.

Le plan du chapitre suit notre démarche : dans un premier temps, nous décrirons le sous-système d'apprentissage perceptif (§3.2). En particulier, nous introduirons l'ensemble des paramètres de la mémoire. Ensuite, nous spécifierons le mécanisme de sélection qui forme l'information perceptive à partir d'un signal et d'une mémoire (§3.3 et §3.4). Enfin, nous décrirons le postulat de rareté de l'information perceptive ainsi que les contraintes appliquées à la mémoire, qui en découlent : la contrainte d'observabilité (CO) et la contrainte d'unicité (CU) (§3.5).

En guise de conclusion à ce chapitre, nous ferons le lien entre notre modèle et des modèles existants. Nous montrerons en particulier qu'il possède un caractère de généricité permettant d'espérer une utilisation dans un large domaine d'applications de traitement du signal.

3.2 Sous-système d'apprentissage perceptif

3.2.1 Caractéristiques de l'information perceptive

La fonction du sous-système d'apprentissage perceptif est identique à celle d'un convertisseur analogique/numérique : à partir d'un signal, il s'agit de délivrer en sortie un élément d'un ensemble défini *a priori*, et qu'on associe à une valeur numérique dans le cas du convertisseur. Dans ce dernier cas, Shannon a établi **des contraintes théoriques** pour que la boucle analogique/numérique/analogique comporte **une perte d'information minimale** : il s'agit des théorèmes d'échantillonnage et de quantification. La notion de **bruit de mesure** découle de cette formalisation. On pourra se référer à l'article fondateur de Shannon [Shannon, 1948] et à une analyse historique du concept d'information [Ségal, 1998].

Notre problématique est différente de celle de Shannon, ce qui nous conduit à introduire une autre notion de l'information, appelée **information perceptive**, qui va être dérivée de **contraintes particulières** construites en fonction de notre postulat fondateur.

Notre approche possède certaines similarités conceptuelles avec celle de la mécanique quantique. Nous en disons quelques mots dans la chapitre 6. Dans notre cas, le sous-système d'apprentissage perceptif est **un détecteur**. Avant l'expérience, nous construisons un ensemble d'événements élémentaires détectables (la mémoire). L'information perceptive est la **conjonction de détections simultanées d'événements élémentaires** au cours de l'expérience.

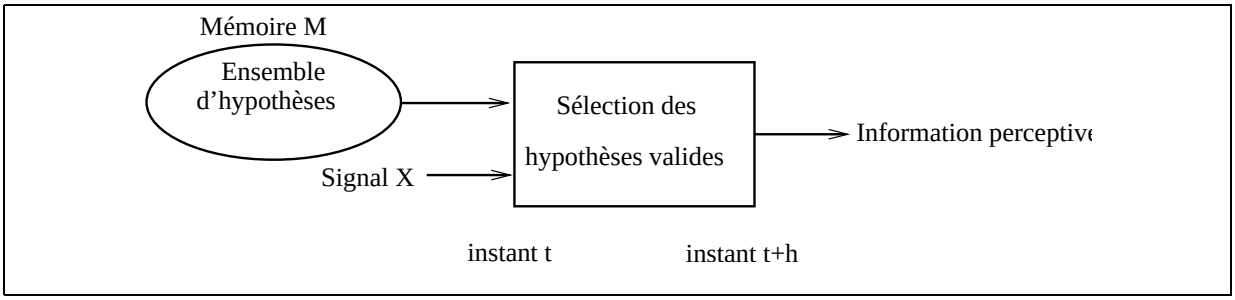


FIG. 3.1 – Vue générale du sous-système d'apprentissage perceptif

Nous caractérisons l'information perceptive, non pas grâce à une mesure sur l'espace du signal (par échantillonnage), mais grâce au calcul **avant l'expérience** de la probabilité de détection des événements élémentaires et à une règle de regroupement de ces événements en informations perceptives (contrainte d'unicité), de laquelle découle la notion d'othogonalité entre deux informations perceptives. Cette règle est basée sur la notion de **simultanéité**.

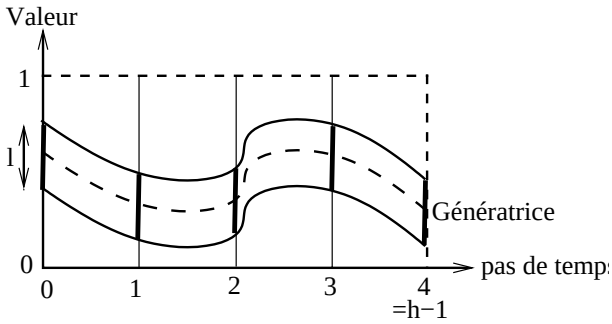
L'espace qui nous intéresse ici est un **espace d'événements**. Chacun des événements élémentaires est relié avec une grandeur physique. Ce lien permet de calculer la probabilité d'apparition de l'événement, qui est à la base des contraintes appliquées à la mémoire. La notion de simultanéité va permettre le regroupement d'événement élémentaires, qui ne sont pas tous forcément associés à la même grandeur physique. Ainsi, **la notion d'information perceptive est, à la base, multi-modale**.

3.2.2 Modèle global

Le schéma du modèle du sous-sytème d'apprentissage perceptif est donné par la figure 3.1. Le mécanisme de sélection possède deux entrées : le signal X, pris entre les instants t et $t+h$, ainsi que la mémoire[1] du système, qui contient un ensemble d'hypothèses d'évolution du signal entre les instant t (début du mécanisme de sélection) et t+h (validation d'une ou de plusieurs hypothèses, le cas échéant[2]). Son rôle est de valider ou d'invalider l'ensemble des hypothèses, prise une à une, contenues dans la mémoire. Nous supposons que ce mécanisme possède une dimension temporelle (la durée de validation h). L'information perceptive est délivrée dans le cas où au moins une hypothèse est validée. Le terme « propriété », que nous avons utilisé dans la sous-section précédente, est représenté ici par une hypothèse d'évolution du signal, faite à l'instant t et validée à l'instant t+h.

Dans la suite de ce document, on normalisera les valeurs du signal X, en supposant qu'elles sont toutes comprises dans l'intervalle [0,1]. On créera le vecteur $(x_t,x_{t+1},...,x_{t+h-1})$ à partir des observations de X faites aux instants t, t+1, ..., t+h-1. Ce vecteur appartient donc à l'hypercube $[0,1]^h$. Les valeurs du vecteur seront récupérées à intervalle de temps constant, qu'on supposera être égal au pas d'échantillonnage associé à la numérisation du signal X.

1. Le terme « mémoire » a été choisi en relation avec le rôle de filtrage que la mémoire biologique semble jouer dans la perception. Ce filtrage se traduit par des capacités d'anticipation qui se mettent en place avec l'expérience. Cela forme actuellement le centre d'un débat scientifique ouvert. 2. Il est possible qu'aucune hypothèse ne soit retenue.



La fonction C(t) est la courbe dessinée en pointillés. Un cylindre, appelé *focus*, est généré à partir de C(t) (génératrice) et du paramètre l (section du cylindre).

FIG. 3.2 – Définition d'un focus

3.2.3 Modèle d'un événement élémentaire

L'ensemble des contraintes qui seront appliquées au sous-système d'apprentissage perceptif se focaliseront sur la structure de la mémoire. Celle-ci est composée d'un ensemble d'hypothèses, qui sont susceptibles d'être détectées au contact de l'environnement du système (c'est-à-dire en récupérant les valeurs d'un signal X).

Une hypothèse est une prédiction sur l'évolution du signal entre les instants t et t+h. Cette prédiction s'effectue à l'instant t. L'hypothèse est composée de deux éléments :

- une fonction C(t) à valeurs dans [0,1]
- un triplet (h,i,l), avec $h \in \mathbb{N}, h \geq 2$, : $i \in \mathbb{N}, i \leq h$, $l \in \mathbb{R}, \in]0,1[$

Le paramètre h représente le nombre de pas de temps sur lesquels on va effectuer une prédiction sur l'évolution du signal. Un cylindre, que nous appellerons **focus**, est constitué à partir du paramètre l (section du cylindre) et de la fonction C(t) (génératrice du cylindre) sur h pas de temps (voir la figure 3.2).

L'événement élémentaire est la détection de la validation de l'hypothèse. Celle-ci sera dite validée au bout de h pas de temps si au plus i valeurs du signal parmi x_t,x_{t+1},...,x_{t+h-1} sont à l'extérieur du cylindre.

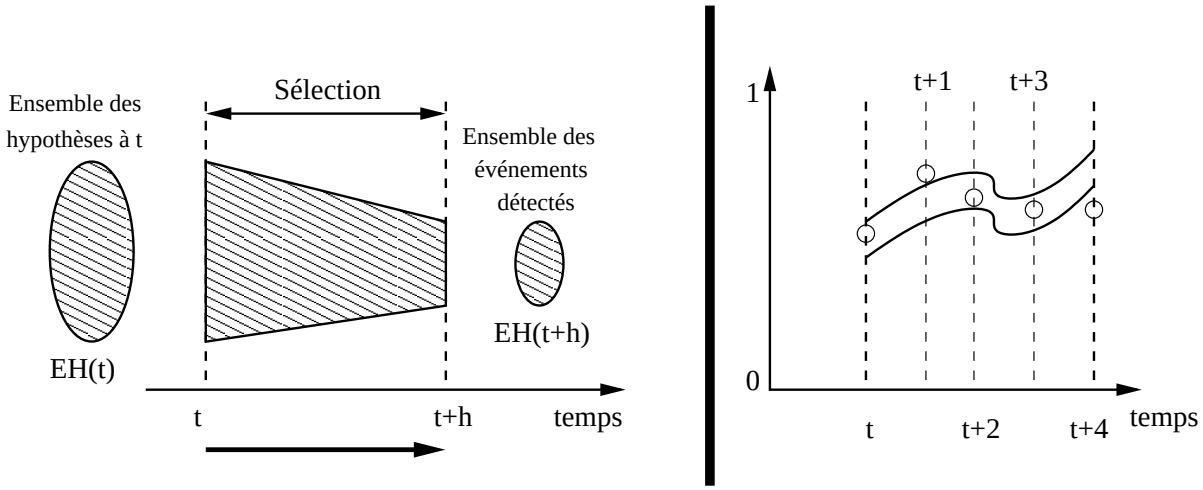
Le paramètre i représente donc un seuil limite d'erreurs à ne pas dépasser pour que l'hypothèse soit validée. La validation de l'hypothèse est donc soumise à deux contraintes, spécifiées *a priori*:

- la durée de validation de l'hypothèse, donnée par le paramètre h
- le taux de valeurs du signal devant être à l'intérieur du focus (désigné par (h-i)/h)

Les contraintes appliquées à la mémoire vont caractériser les triplets (h,i,l) valides, en fonction de l'ensemble des hypothèses (triplet + focus) composant la mémoire.

3.2.4 Mécanisme de sélection des hypothèses valides

Le mécanisme de sélection (voir la figure 3.3) va consister à considérer, **à chaque instant t**, l'ensemble des hypothèses contenues en mémoire et à les valider ou les invalider suivant l'adéquation du signal avec chaque hypothèse (triplet (h,i,l) et focus). Cette adéquation se mesure en



La sélection (figure de gauche) élimine, au cours du temps, l'ensemble des hypothèses n'ayant pas respecté leur contrainte. La figure de droite montre la confrontation d'une hypothèse ayant comme paramètres h=5,i=1 et l=0.1. Dans ce cas, 3 points sur 5 sont situés à l'intérieur du cylindre associé à l'hypothèse : les contraintes ne sont donc pas respectées, car le nombre de valeurs du signal à l'extérieur du cylindre est 2, supérieur strictement à i.

Fig. 3.3 – Sélection d'une hypothèse.

comptant le nombre de valeurs du signal dans chaque focus ; ce nombre doit être supérieur à $h-i$ pour que l'hypothèse soit validée. La durée de validation dépendra de chacune des hypothèses (elle est fixée par le paramètre h propre à chaque hypothèse).

L'ensemble des hypothèses validées au cours du temps constituera l'information qui sera délivrée au système d'atteinte d'objectif. Il faut bien noter qu'à un instant donné, il se peut qu'aucune hypothèse ne soit validée ou que, au contraire, plusieurs hypothèses soient validées.

3.3 Mécanisme de sélection pour un ensemble fini d'hypothèses

3.3.1 Introduction

Nous avons présenté les composantes du sous-système d'apprentissage perceptif : il s'agit d'un mécanisme de sélection faisant appel au signal X pour valider ou invalider un ensemble d'hypothèses pré-établies (la mémoire). À chaque instant, on suppose que chaque hypothèse de la mémoire est susceptible d'être validée : la sélection va permettre d'éliminer l'ensemble des hypothèses ne satisfaisant pas leur contrainte[1].

Dans le cas où le nombre d'hypothèses contenues dans la mémoire est fini, le mécanisme de sélection s'exprime à l'aide d'un algorithme très simple : c'est l'objet de cette section.

1. Pour qu'une hypothèse soit validée, il faut que, au moins, h-i valeurs du signal se trouvent dans le focus associé à celle-ci.

3.3.2 Algorithme

L'ensemble des hypothèses sera noté M = M_1,M_2,...,M_n [1]. Une hypothèse M_j est associée à une fonction génératrice $C_j(t)$, ainsi qu'au triplet (h_j,i_j,l_j). L'algorithme de sélection peut être rapproché d'un **problème d'exploration/exploitation sur une période de temps finie**. En effet, il possède deux fonctions antagonistes qui s'exercent à chaque instant t : une génération d'hypothèses, qui tend à augmenter le nombre d'hypothèses qui doivent être testées à chaque instant, et une élimination d'hypothèses. Nous allons expliciter ces deux parties.

La génération d'hypothèses est très simple. Elle a pour but **d'explorer** l'ensemble des possibilités d'évolution du signal contenues dans la mémoire. Pour cela, à **chaque instant t**, on génère l'ensemble total des hypothèses, noté E(t) : E(t) = M. On ajoute cet ensemble à l'ensemble, noté TE(t), des ensembles E(t_k), pour $t_k \leq t$, coexistant à cet instant t. Par conséquent, à chaque instant t, l'ensemble total des hypothèses considérées est augmenté du cardinal de M, c'est-à-dire n.

Voici comment fonctionne l'élimination d'hypothèses. À chaque instant t, on récupère la valeur du signal X(t) et on regarde, pour chacune des hypothèses de TE(t) [2], si cette valeur appartient au focus associé à cette hypothèse. Si ce n'est pas le cas, on incrémente un compteur d'erreur, $K(t_k,j)$, qui est propre à l'hypothèse M_j de l'ensemble E(t_k). Ainsi, on va pouvoir éliminer des éléments M_j de E(t), pour deux raisons : soit M_j ne respecte pas sa contrainte ($K(t_k,j) > i_j$), soit la durée de validation h_j est atteinte ($t - t_k = h_j$) et l'hypothèse M_j, formulée à un instant $t_k < t$, est validée. Lorsqu'une hypothèse, formulée à un instant t_k est soit validée, soit invalidée, on la retire de l'ensemble $E(t_k)$. Si cet ensemble devient vide (on a retiré l'ensemble des hypothèses), on retire l'ensemble $E(t_k)$ de l'ensemble TE(t). La procédure d'élimination permet de conserver un ensemble réduits d'hypothèses qui traduisent l'évolution réelle du signal. Ces hypothèses pourront être **exploitées** pour former une information perceptive.

Lorsqu'une hypothèse est validée, on l'ajoute à l'ensemble S(t) qui contient les hypothèses validées à l'instant t (et uniquement à l'instant t).

Les deux processus de génération et d'élimination d'hypothèses forment l'algorithme de sélection 3.1.

3.3.3 Précisions concernant l'algorithme de sélection

Le nombre d'hypothèses à valider peut-il croître indéfiniment ? Il n'est pas difficile de montrer que ce nombre est majoré. Pour cela, il suffit de noter que le nombre d'ensembles $E(t_k)$, présents à l'instant t dans l'ensemble TE(t), est majoré. En effet, la durée d'existence d'un ensemble $E(t_k)$ est majorée par la durée maximum de validation d'une hypothèse de M, c'est-à-dire $\max_{j\in\{1,...,n\}}\{h_j\}$. Par conséquent, le nombre maximum d'ensembles $E(t_k)$ présents dans $TE(t)$ est majoré par $\max_{j\in\{1,...,n\}}\{h_j\}$. On en déduit que le nombre d'hypothèses à valider est majoré par $n.\max_{j\in\{1,...,n\}}\{h_j\}$. Cependant, on se rend compte que, même si cette limite est bornée, la borne supérieure peut devenir rapidement un **frein à des capacités d'utilisation en temps réel**. En effet, les tests sur l'appartenance d'une valeur du signal X(t) à chacune des hypothèses

1. La notation M_j n'est, bien entendu, rien à voir avec celle employée pour les marquages du graphe d'état dans la première partie de ce document de thèse. 2. C'est-à-dire chacune des hypothèses de chacun des $E(t_k)$ de TE(t).

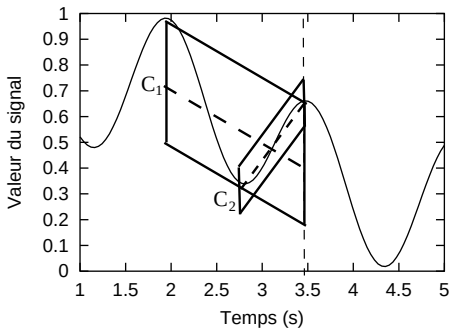
Algorithme 3.1 Algorithme de sélection pour un ensemble fini M d'hypothèses

Pour chaque instant t, faire
    E(t) := M
    Ajouter l'ensemble E(t) à TE(t) et initialiser l'ensemble des compteurs K(t,j) à 0
    Récupérer la valeur de X(t)
    S(t) := ∅
    Pour chacun des ensembles E(t_k) de TE(t), faire
        Pour chacune des hypothèses M_j de E(t_k), faire
            Si X(t) ∉ [C_j(t - t_k) - l_j/2, C_j(t - t_k) + l_j/2,], alors
                K(t_k,j) := K(t_k,j) + 1
            FinSi
            Si (K(t_k,j) >i_j, alors
                l'hypothèse M_j est invalidée et on retire M_j de E(t_k)
            FinSi
            Si (t - t_k = h_j), alors
                l'hypothèse M_j est validée : on retire M_j de E(t_k) et on l'ajoute à S(t)
            FinSi
        FinPour
    FinPour
    L'ensemble S(t) est transmis à la sortie du processus de sélection
FinPour

s'effectuent **à chaque pas de temps**.

La sortie S(t) du processus de sélection peut être vide. Lorsqu'on initialise le système, elle est obligatoirement vide pendant les h_{min} premiers pas de temps (avec $h_{min} = \min_{j\in\{1,...,n\}}\{h_j\}$). D'autre part, S(t) fournit une information sur l'évolution passée du signal X, sur des périodes de temps qui peuvent être différentes (qui dépendent de la valeurs de h_j). Ainsi, la sortie S(t) peut fournir **simultanément** des informations à partir d'hypothèses ayant été générées à des instants t_k différents.

L'algorithme 3.1 donne, à chaque instant t, un ensemble d'hypothèses validées. Ces hypothèses sont associées à des focus, qui décrivent l'évolution du signal entre un instant $t_k < t$ et l'instant t. On ne peut donc pas reconstruire le signal à partir de S(t) : une hypothèse M_j indique une localisation vague (dépendant de la largeur l_j de chaque focus C_j) du vecteur (x_{t_k},x_{t_k+1},...,x_t dans l'espace $[0,1]^{h_j}$. L'ensemble S(t) est donc une superposition, à des échelles de temps h_j différentes, de ces localisations. La figure 3.4 illustre nos propos. Dans cet exemple, S(t) est composé de deux hypothèses possédant des échelles de temps différentes et traduisant deux évolutions opposées du signal : l'une est « descendante » (génératrice C_1) et l'autre est « ascendante » (génératrice C_2). L'information perceptive issue de ce signal comporte donc des composantes liées chacune à une échelle de temps particulière.



La figure montre l'ensemble S(t) à l'instant t=3.5 s, qui est composé de deux hypothèses, dont les focus sont représentés par les deux parallélogrammes en trait épais, de génératrices C_1 et C_2. Ces deux hypothèses ont été générées aux instant t=2 s et t=2.7 s .

FIG. 3.4 – Exemple d'un ensemble S(t) comportant deux hypothèses

3.4 Mécanisme de sélection pour un ensemble infini d'hypothèses

3.4.1 Introduction

Dans la section précédente, nous avons développé un algorithme traitant du cas où l'ensemble des hypothèses est fini. On peut envisager que l'ensemble des hypothèses soit infini. Dans ce cas, l'algorithme précédent ne s'applique plus. Dans cette section, nous nous limitons aux cas où l'ensemble des fonctions génératrices C(t) s'expriment en fonction de paramètres continus. Nous montrerons alors qu'il est possible de trouver un algorithme, utilisant le calcul sur les intervalles, qui permet d'encadrer l'ensemble des hypothèses valides d'une manière fiable à l'aide de deux ensembles : l'un contenant l'ensemble des solutions et l'autre étant inclus dans celui-ci.

3.4.2 Constitution de la mémoire - notations

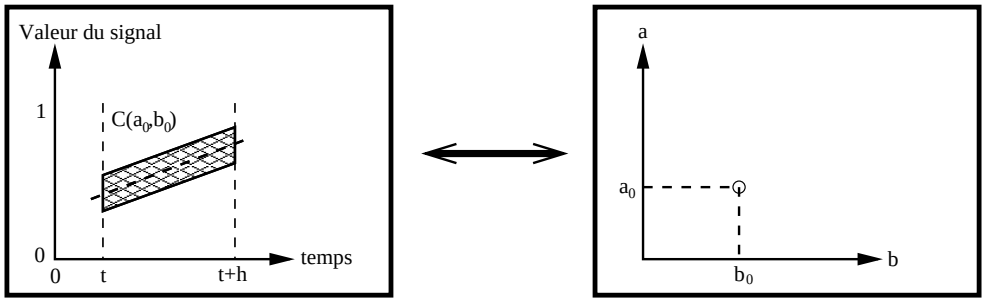
Nous allons décrire une catégorie de mémoire particulière, possédant un ensemble infini d'hypothèses. Ce cas est intéressant, car nous allons montrer qu'on peut trouver une approximation de l'ensemble S(t) à chaque pas de temps.

Nous supposons que l'ensemble des hypothèses sont associées à un **unique** triplet (h,i,l). Par contre, chaque hypothèse est reliée à un focus dont la génératrice est une fonction paramétrique C à p paramètres réels a_1,a_2,...,a_p. Les valeurs de ces p paramètres constituent un vecteur α, de dimension p : $\alpha = (a_1,a_2,...,a_p)$. La valeur de la génératrice à l'instant t sera notée $C_\alpha(t)$.

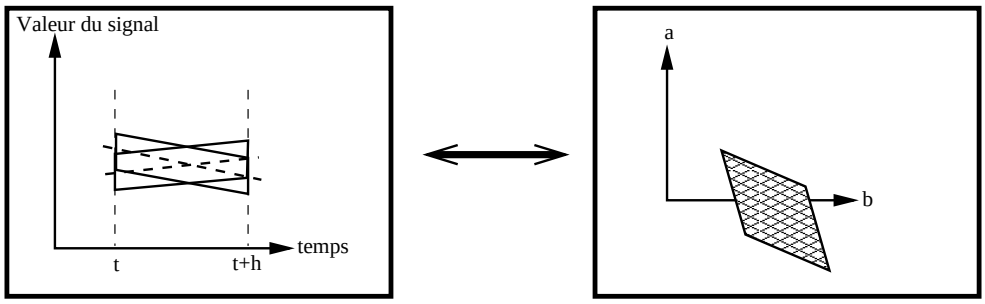
Nous supposerons que la génératrice d'un focus ne peut être associée qu'à un unique vecteur α. Dans ce cas, une hypothèse est entièrement décrite par la connaissance de la valeur de α [1,2]. Le graphe (a) de la figure 3.5 montre cette correspondance dans le cas où *alpha* est de dimension 2 (les deux dimensions sont notées a et b sur le graphe). Ainsi, l'ensemble des solutions à l'instant t, noté S(t), qui est composé des hypothèses validées à t, pourra être associé à l'ensemble des vecteurs α. Le graphe (b) de la figure 3.5 montre l'expression d'un ensemble de solutions dans l'espace des α.

1. En effet, le triplet (h,i,l) est commun à l'ensemble des hypothèses et la valeur de α est représentative de la génératrice du focus associé à une hypothèse particulière. 2. On voit ici les liens avec la transformée de Hough.

(a) un unique focus



(b) un ensemble de focus généré à partir de deux paramètres a et b



Les graphes temps/signal sont donnés à gauche, alors que les graphes de droite représentent l'ensemble des valeurs de α correspondant aux focus dessinés sur les graphes de gauche. Les génératices sont des droites, déterminées à partir de deux paramètres a et b (pente et valeur à l'origine). Pour la figure (b), nous ne représentons les deux focus « extrêmes » à gauche ; l'ensemble des valeurs de α associées à ces focus sont données, à droite, par le parallélogramme hachuré.

FIG. 3.5 – Transformée de Hough.

3.4.3 Formalisation de l'ensemble S(t) des solutions - résolution d'un problème d'inversion ensembliste

Nous sommes dans le cas où un triplet (h,i,l) est fixé pour l'ensemble des hypothèses, les génératrices étant paramétrées par le vecteur α. Le problème que nous avons à résoudre pour déterminer l'ensemble S(t) des solutions, à l'instant t, revient à résoudre un problème inverse : trouver l'ensemble des vecteurs α tels que au moins h-i relations parmi h possibles soient satisfaites dans le systèmes 3.1.

$$\left\{\begin{array}{c} (I_D)\ \forall t \in [0,h-1], C_\alpha(t) \in [0,1] \\ (I_1)\ x_{t-h+1} \in [C_\alpha(0) - l/2, C_\alpha(0) + l/2] \\ (I_2)\ x_{t-h+2} \in [C_\alpha(1) - l/2, C_\alpha(1) + l/2] \\ \vdots \\ (I_k)\ x_{t-h+k} \in [C_\alpha(k-1) - l/2, C_\alpha(k-1) + l/2] \\ \vdots \\ (I_h)\ x_t \in [C_\alpha(h-1) - l/2, C_\alpha(h-1) + l/2] \end{array}\right. \tag{3.1}$$

La relation I_D doit être impérativement satisfaite. Elle exprime le fait que les génératrices doivent avoir des valeurs comprises entre 0 et 1.

3.4.4 Méthode de résolution

Le système 3.1 est très général. Dans le cas où les fonctions génératrices sont des droites, possédant deux paramètres a et b (la valeur à l'origine), $\alpha = (a/h, b)$. On paramètre chaque génératrice par l'équation $C_\alpha(t) = \frac{a}{h}t + b$. Le système devient alors :

$$\left\{\begin{array}{c} (I_D)\ b \in [0,1],\ a + b \in [0,1] \\ (I_1)\ x_{t-h+1} - l/2 \leq b \leq x_{t-h+1} + l/2 \\ (I_2)\ \frac{h}{k}(x_{t-h+2} - \frac{l}{2} - b) \leq a \leq \frac{h}{k}(x_{t-h+2} + \frac{l}{2} - b) \\ \vdots \\ (I_k)\ \frac{h}{k}(x_{t-h+k} - \frac{l}{2} - b) \leq a \leq \frac{h}{k}(x_{t-h+k} + \frac{l}{2} - b) \\ \vdots \\ (I_h)\ x_t - \frac{l}{2} - b \leq a \leq x_t + \frac{l}{2} - b \end{array}\right. \tag{3.2}$$

Lorsque i=0 (on désire trouver a et b tels que toutes les inégalités soient satisfaites), on peut résoudre le problème par **la méthode du simplexe**. Mais, lorsque i est non nul, il n'est pas possible d'appliquer cette méthode qui donne une solution respectant l'intégralité des inéquations. La solution peut être obtenue en considérant l'union des solutions apportées à des systèmes de h-k inéquations, pour k variant de 0 à i : le nombre de système à résoudre est donc égal à $\sum_{k=0}^{i} C_{h,k}$, qui peut être **très important**. D'autre part, la méthode du simplexe ne s'applique que si l'expression des C_α est **linéaire**. Pour toutes ces raisons, il nous faut une autre méthode.

La première solution est d'effectuer un maillage du domaine des vecteurs α valides, indiqué par la relation (I_D). Pour chaque point de ce maillage, on peut alors vérifier le nombre de relations satisfaites et déduire de cela l'ensemble des points satisfaisant nos contraintes. Mais, cette solution présente des inconvénients : il est difficile *a priori* de connaître la grosseur du maillage à appliquer : si celui-ci est trop fin, le nombre de points à tester explose suivant le nombre de dimensions du vecteur α, mais si celui-ci est trop grossier, on risque de perdre des solutions.

Il existe une méthode qui est particulièrement bien adaptée à notre problématique : il s'agit d'utiliser les propriétés du calcul sur les intervalles[1] et d'exprimer l'équation 3.1 sous la forme d'un **problème d'inversion ensembliste**, qui peut être mis sous la forme générique suivante :

$$y = f^{-1}(x)$$

Le principe de résolution du problème d'inversion ensembliste, lié au système 3.1, est décrit dans [Jaulin et Walter, 1993] ou dans [Jaulin et al., 1996]. Dans notre cas, il s'agit de découper récursivement l'ensemble initial y en boîtes pour lesquelles on va calculer l'image par f. Voici comment nous choisissons x, y et f :

- y est une boîte incluant l'ensemble des valeurs de *alpha* vérifiant la relation (I_D)
- x est un intervalle inclus dans [0,h]
- f(y) est un intervalle [min,max] défini de la manière suivante : min est égal au plus petit nombre de relations de 3.1 satisfaites en considérant l'ensemble des éléments de y, alors que max est le plus grand nombre de relations satisfaites en considérant l'ensemble des éléments de y.

Pour calculer f(y), il faut pouvoir inverser chaque relation (I_k), de manière à la mettre sous la forme : $\alpha \in G(x_{t-h+k},l)$, où G est une fonction à valeurs dans $\mathbb{R}^p$. On associe à (I_k) la valeur 1 si $y \subset G(x_{t-h+k},l)$, la valeur 0 si $y \bigcap G(x_{t-h+k},l) = \emptyset$ et l'intervalle [0,1] sinon. La somme de ces valeurs donne un intervalle inclus dans [0,h].

La théorie nous assure qu'on peut classer les boîtes y en trois catégories :

1. les boîtes pour lesquelles **on est certain** que l'intégralité des points de la boîte vérifient nos contraintes (au moins h-i relations satisfaites) : l'image par f de ces boîtes donne un intervalle [min,max] tel que min $\geq$ h-i.
2. les boîtes pour lesquelles **on est certain** que l'intégralité des points de la boîte ne vérifient pas nos contraintes (plus de i relations ne sont pas satisfaites) : l'image par f de ces boîtes donne un intervalle [min,max] tel que max $<$h-i.
3. les boîtes pour lesquelles il existe à la fois des points vérifiant les contraintes et des points ne vérifiant pas les contraintes : l'image par f de ces boîtes donne un intervalle [min,max] tel que min $<$h-i et max $\geq$ h-i.

Le découpage s'effectue sur les boîtes appartenant à la troisième catégorie. Un critère d'arrêt sur la dimension minimale d'une boîte permet de maîtriser la précision de l'ensemble S(t) des solutions.

Les quatre points particulièrement intéressants de cette méthode sont les suivants :

- elle peut gérer les problèmes pour lesquels h-i relations parmi h doivent être satisfaites
- elle autorise l'utilisation de génératrices C_α non linéaires
- elle découvre l'intégralité des solutions, même lorsque l'ensemble des solutions est formé de parties non connexes
- elle permet d'encadrer l'ensemble S(t) par deux ensembles : l'un est représenté par l'union des boîtes de première catégorie et l'autre est formé par l'union des boîtes de la 1ère et de la troisième catégorie. Le premier ensemble est inclus dans S(t), alors que le deuxième contient S(t).

1. Le lecteur pourra consulter [Moore, 1979] pour avoir une référence sur ce domaine scientifique.

Cependant, lorsque la dimension des vecteurs α n'est pas petite (en pratique, inférieure à 5 environ), le nombre de boîtes créées peut s'accroître énormément, rendant impossible l'inclusion du calcul de S(t) dans un processus fonctionnant en temps réel. Cependant, nous avons constaté que, pour une dimension de α valant 2, la méthode présentée ci-dessus est entre 2 et 4 fois plus rapide qu'avec une méthode utilisant un maillage comportant 10.000 points [1].

3.4.5 Algorithme

Nous utilisons la méthode par découpage d'intervalles. L'algorithme 3.2 que nous présentons ici s'applique à un problème pour lequel les génératrices sont des fonctions paramétriques, comportant p paramètres. Nous rappelons que les hypothèses composant la mémoire sont associées à un unique triplet (h,i,l).

Cet algorithme découle directement de l'algorithme SIVIA [2] développé par Jaulin et Walter. La fonction G est celle décrite dans la sous-section précédente. Dans la fonction CalculF() décrite dans l'algorithme, y désigne une boîte incluse dans $\mathbb{R}^p$, dont les composantes suivant les différents axes sont les intervalles y_1,y_2,...,y_p. Dans l'algorithme principal, y_{init} désigne la boîte initiale, qui va être successivement découpée grâce à la fonction SIVIA(). Celle-ci retourne deux ensembles : le premier est inclus dans l'ensemble S(t) et l'autre contient S(t) [3]. La fonction Decoupe() découpe une boîte y en deux boîtes de même volume, selon l'axe portant l'intervalle y_k dont la longueur est la plus grande. Les fonctions min et max désignent respectivement la borne inférieure et la borne supérieure d'un intervalle. La fonction Largeur appliquée à une boîte y désigne la longueur du plus petit intervalle y_k. Le critère de divisibilité d'une boîte y est désigné par le paramètre δ.

3.5 Contraintes appliquées à la mémoire

3.5.1 Introduction

Nous allons exposer dans cette section les idées qui permettront d'établir les contraintes qui vont s'appliquer à la mémoire du système : la contrainte d'observabilité (CO) et la contrainte d'unicité (CU). La contrainte CO se base sur l'idée que l'information perceptive est rare. Nous reviendrons donc, tout d'abord, sur ce point. Nous donnerons ensuite l'énoncé des deux contraintes.

La contrainte CO est déduite des résultats d'une étude qui a été notre point de départ pour la construction de l'information perceptive. Cette étude utilise un modèle moins général de l'information perceptive, qui ne correspond plus avec celle que nous utilisons actuellement. Cependant, le lecteur pourra y trouver une partie de notre cheminement intellectuel aboutissant au concept d'information perceptive présenté dans le corps du document. Nous la fournissons donc en annexe B.1. En particulier, **on y déduit empiriquement un premier résultat comparable au théorème de Shannon sur l'échantillonnage**.

3.5.2 Notion d'événement rare

Pour comprendre notre notion de rareté, il faut noter la différence entre les deux significations de la probabilité (voir Carnap, par exemple [Carnap, 1950]). Il existe **une probabilité** *a posteriori*, qui est déterminée par le rapport entre le comptage du nombre d'occurrences d'un

1. Ce nombre est un minimum si on souhaite ne pas « oublier » de solutions. 2. Set Inverter for Interval Analysis 3. C'est le sens donné à « Retourner ensemble1 et ensemble2 ».

Algorithme 3.2 Algorithme de sélection pour un ensemble infini M d'hypothèses, dont les génératrices sont paramétrées par p paramètres

Fonction CalculF(boîte y)
Début
    Intrevalle F
    Pour k allant de 1 à h, faire
        Si $y \subset G(x_{t-h+k},l)$, alors
            F := F + 1
        Sinon Si $y \bigcap G(x_{t-h+k},l) \neq \emptyset$ alors,
            F := F + [0,1]
        FinSi
    FinPour
    Retourner S
Fin

Fonction SIVIA(boîte y)
Début
    Si min(CalculF(y)) $\geq$ h-i, alors
        Retourner y et y
    Sinon si max(CalculF(y)) <h-i, alors
        Retourner $\emptyset$ et $\emptyset$
    Sinon si Largeur(y) $>\delta$, alors
        (y_a,y_b) = Decoupe(y)
        Retourner SIVIA(y_a) $\bigcup$ SIVIA(y_b) et SIVIA(y_a) $\bigcup$ SIVIA(y_b)
    Sinon,
        Retourner $\emptyset$ et y
    FinSi
Fin

Algorithme principal
Début
    Pour chaque instant t, faire
    Récupérer la valeur de X(t)
    (inclus, contenant) := SIVIA(y_{init})
    Les deux ensembles encadrant S(t) sont transmis à la sortie du processus de sélection
Fin

événement sur le nombre total d'occurrences : il s'agit d'une *statistique*, effectuée en utilisant l'expérience réelle. Mais, en physique notamment, il existe également **une probabilité** *a priori*, dite *objective*, qui est une propriété à part entière d'un événement, qu'on peut déterminer par le calcul sans faire appel à l'expérience réelle. Voici un exemple, qui permettra d'imager ce que nous appelons « rareté ».

Considérons une pièce rectangulaire étanche, remplie d'air. On peut dire que chaque molécule de gaz contenue dans cette pièce est identifiable à une « bille » qui se déplace librement avec une trajectoire rectiligne uniforme, jusqu'à ce qu'elle percute une autre bille, provoquant un changement de direction de la trajectoire. Cette modélisation est suffisamment réaliste pour permettre de déterminer statistiquement certaines propriétés macroscopiques des gaz (loi des gaz parfaits, par exemple). Mais, elle permet également d'envisager certains événements qui nous semblent impossibles *priori*. En voici un : l'ensemble des molécules de gaz se trouvent dans la même moitié de la pièce pendant une durée déterminée. Cet événement serait, bien évidemment, particulièrement fâcheux, du moins si une personne se trouvait à ce moment dans la partie vide de gaz. La probabilité *objective* de celui-ci se calcule et est non nulle. Cependant, la probabilité *statistique* de celui-ci est nulle : on n'a jamais rapporté l'occurrence d'un tel événement. Bien entendu, si on attendait un temps infini, celui-ci se produirait. Mais, dans une durée d'expérience rapportée à l'échelle « humaine », cet événement ne se produit pas. Voilà comment nous définissons notre notion de « rareté ».

Il existe pourtant **une différence théorique** majeure entre un événement impossible et un événement rare : on ne pourra jamais prouver que ce dernier est impossible. Cependant, il n'existera **aucune différence en réalité** : aucun de ces deux événements ne se produira, du moins sur une durée d'expérience finie et « raisonnable »[1].

Notre idée est donc de centrer nos efforts de formalisation autour de la maîtrise de la probabilité d'occurrence d'événements fâcheux. En autorisant une probabilité d'erreur non nulle, nous espérons pouvoir restreindre les contraintes dont il faudrait se munir pour établir des preuves[2].

3.5.3 Le problème « D »

Nous souhaitons expliquer l'importance du paramètre h de la mémoire du système. Ce paramètre influe sur la dimensionnalité théorique de l'ensemble des signaux possibles : un signal pris entre les instants t et t+h permet d'engendrer un vecteur de dimension h (en normalisant le pas d'échantillonnage à 1). **L'idée est que l'information perceptive découverte est fiable uniquement si la probabilité de découverte au hasard d'une telle information est très faible**. Dans ces conditions, le guide permettant de découvrir l'information perceptive est l'expérience du système, emmagasinée dans sa mémoire. Nous illustrons cela par l'exemple nommé **problème « D »**[3]. Celui-ci va expliciter la caractéristique de rareté de l'information perceptive et son lien avec la mémoire. Il met en scène une personne cobaye, qui doit donner des réponses à des questions posées, et un observateur qui ne connaît pas la nature du problème, mais sait si la personne cobaye a bien répondu ou non. C'est l'observateur qui nous intéresse : il doit décider, à

1. Il est important de noter que la notion de rareté est associée avec une expérience en temps limité. La valeur de la probabilité que nous pourrons associée avec l'occurrence d'un événement rare dépend à la fois de cette durée et du degré de confiance qu'on souhaite avoir sur la non obtention de cet événement. 2. La théorie de la transmission du signal utilise avec succès cette démarche intellectuelle. Voir [Shannon, 1948]. 3. Le terme « D » vient de la nature « Décisionnelle » du problème de l'observateur.

partir des résultats de la personne cobaye, si **celle-ci a répondu au hasard ou en utilisant ses connaissances sur le problème posé**.

Voici l'énoncé du problème « D ».

Considérons les deux classes de problèmes suivantes :

1. Imaginons que nous jetions une pièce de monnaie en l'air à l'instant t. Considérons l'ensemble des événements possibles générés par ce jet à l'instant t+1 : « la pièce tombe sur pile » et la « pièce tombe sur face ». *A priori*, si la pièce est correctement équilibrée, ces deux événements sont statistiquement aussi probables l'un que l'autre. Imaginons à présent une personne jetant cette pièce à l'instant t, après avoir effectué une prévision sur le résultat du jet, à l'instant t-1. D'un point de vue statistique, le taux de bonnes prédictions sur un nombre d'essais très grand correspond à la probabilité objective que la pièce tombe sur pile ou sur face. Ainsi, si la personne effectue non pas un jet mais disons 100 jets, après avoir donné une prédiction sur l'enchaînement précis des 100 jets, la probabilité de justesse de l'intégralité de la prédiction est égale à $(1/2)^{100} \simeq 8.10^{-31}$. Cette très faible probabilité de deviner la bonne réponse est due à la très grande dimension de l'univers des possibilités qui sont offertes à la personne lors de son choix (dans notre exemple, il y en a exactement $2^{100} \simeq 10^{30}$). D'autre part, il paraît raisonnable d'affirmer que la personne servant de cobaye à l'expérience n'influence pas le résultat de celle-ci. Sachant cela, que penserions-nous si cette personne avait effectivement deviné l'enchaînement exact des jets, malgré cette si faible probabilité de bonne réponse?
2. Considérons à présent une personne devant répondre à un questionnaire comportant 100 questions. Admettons que celle-ci ait deux possibilités de réponses : « vrai » ou « faux » et que les questions soient toutes du type « Le nombre entier 'n' est pair ». Pour déterminer les 100 entiers du questionnaire, on les choisit entièrement au hasard, en imposant simplement qu'un entier ne soit choisi qu'une fois au plus. Si on considère la personne comme un acteur passif de l'expérience [1], la probabilité objective pour que les réponses au questionnaire soient toutes exactes est la même que celle de l'expérience précédente, puisqu'il existe une chance sur deux de répondre correctement à chaque question.

Les deux expériences sont objectivement similaires et possèdent deux univers des possibilités de même dimension. Pourtant, un élément essentiel diffère : dans le premier cas, on peut raisonnablement supposer que la personne cobaye ne maîtrise pas le résultat de son jet, donc que ses prévisions sont purement spéculatives, alors que dans le second cas, si la personne cobaye possède la notion de nombre pair ou impair, elle saura répondre avec certitude à l'intégralité du questionnaire. Mais comment discerner en pratique ces deux cas de figures? Pour cela, admettons qu'une deuxième personne (l'observateur) puisse choisir *a priori* l'étendue de l'univers des possibilités offertes à la personne cobaye, commun aux deux expériences, sans pour autant en connaître la nature : elle est avertie uniquement de la qualité des résultats de la personne cobaye pour chacune des deux tâches, sans savoir en quoi celles-ci consistent, mais en maîtrisant toutefois la probabilité de bonne réponse. Dans notre exemple, cela signifie que l'observateur donne *a priori* le nombre de jets, ainsi que le nombre de questions (qui sont identiques dans ce cas précis). Une technique simple permettant de discerner les deux expériences est de regarder l'évolution des bonnes réponses en fonction d'une augmentation progressive de la taille de l'univers des possibilités. On imagine facilement que, pour la première expérience, une demande de prévision sur l'enchaînement d'un nombre de jets trop important n'aboutira à aucune bonne réponse. En

1. C'est-à-dire qu'il intervient, comme dans l'expérience précédente, uniquement pour choisir une réponse au hasard

ce qui concerne la deuxième expérience, le même résultat sera obtenu si la personne cobaye ne connaît pas la notion de nombre pair, alors qu'un résultat invariablement positif sera obtenu dans le cas contraire, quel que soit le nombre des questions. Nous souhaitons fixer h de manière à ce que les problèmes de catégorie 1 ne puissent être résolus que *rarement*. Ainsi, lorsque h augmente, le taux de bonnes réponses du cobaye diminue pour les problèmes de catégorie 1 mais reste très bon pour les problèmes de catégorie 2. Si on considère un observateur ne connaissant pas le problème du cobaye, mais décidant du nombre h, celui-ci peut déterminer, simplement en faisant augmenter h, quelle est la catégorie du problème que traite le cobaye. Il faut bien noter que cette réponse est dépendante des capacités du cobaye, si celui-ci est confronté à un problème de catégorie 2 qu'il ne sait pas résoudre : en effet, le cobaye donnerait alors des réponses aléatoires, comme s'il avait été confronté à un problème de catégorie 1.

Ainsi, si on augmente suffisamment la valeur de h, nous savons que la probabilité pour que le cobaye trouve une bonne réponse pour un problème de catégorie 1 est très faible (cet événement est rare). Nous utilisons alors un **raisonnement inductif** pour supposer que, si la personne cobaye donne une bonne réponse alors que h est suffisamment élevé, cela signifie que le problème ne peut pas être de catégorie 1. Dans ce cas, **une unique bonne réponse** suffit pour dire que le problème est de catégorie 2.

Voici comment nous pouvons interpréter le problème « D ». L'observateur est le sous-système d'apprentissage d'objectif. Le choix du cobaye est assimilable à une **hypothèse** contenue dans la mémoire en entrée du processus de sélection. Dans notre cas, h vaudrait 100 et on souhaiterait qu'il y ait une bonne réponse sur l'intégralité des questions, ce qui signifie que i vaudrait 0 (aucune erreur admise). Enfin, un seul choix est proposé au cobaye, ce qui peut être interprété par le fait que la mémoire en entrée du processus de sélection est composée d'une unique hypothèse.

3.5.4 Extensions du problème « D »

Les problèmes utilisant la perception n'appartiennent jamais à une des deux catégories citées dans le problème « D » : il existe toujours une part d'aléas. Voici comment nous pouvons étendre la définition de la catégorie 2 pour obtenir un résultat similaire à celui du problème « D » **dans les cas où aucun problème posé au cobaye n'est totalement déterministe**. Il suffit, pour cela, de donner au cobaye la possibilité de se tromper sur un certain nombre de réponses parmi les h possibles : c'est la signification du paramètre i. On sent bien que les problèmes de catégorie 1 ne seront jamais résolus si h est assez grand et i raisonnablement petit (par rapport à h). Nous utilisons alors un raisonnement inductif pour décréter que, si la probabilité que le cobaye donne une bonne réponse au hasard (malgré les i erreurs qui lui sont accordées) est très faible, et que ce dernier donne néanmoins au moins $h-i$ bonnes réponses, alors c'est un problème de catégorie 2 qui est traité.

Nous pouvons encore donner plus de chances à notre cobaye : nous allons l'autoriser non seulement à commettre au plus i erreurs mais également à donner plusieurs grilles de réponses (chacune comportant h réponses). Pour chacune d'entre-elles, i erreurs sont tolérées. **Cet ensemble de réponses possible pourra être comparé à l'existence de plusieurs hypothèses au sein de la mémoire**.

Mais nous voyons apparaître un problème : reprenons le cas du jet de pièce, qui est de catégorie 1. Admettons que le cobaye doive prévoir la succession des pile et face sur 100 jets, avec

aucune possibilité d'erreur sur les 100 jets (i=0). Que se passe-t-il si on ne restreint pas le nombre de grilles que le cobaye peut remplir ? Il pourra en livrer 2^{100}, comportant chacune une combinaison différente, ce qui lui permettra de trouver la bonne réponse à coup sûr. L'observateur en déduira à tort que le problème du cobaye est de catégorie 2. Il est donc nécessaire de limiter le nombre d'hypothèses possibles, de manière à conserver la possibilité d'avoir une probabilité de bonne réponse très faible. Le raisonnement inductif peut alors s'appliquer pour détecter un problème de catégorie 2.

Par conséquent, les paramètres qui influent sur la fiabilité de la décision de l'observateur (ou du sous-systsème d'apprentissage perceptif) sont les composantes de la mémoire en entrée de ce processus : les paramètres h, i et l (l est en rapport avec la probabilité pour que le cobaye donne une unique bonne réponse au hasard : il vaudrait 1/2 dans le cas des deux exemples du problème « D »), ainsi que le nombre et la nature des hypothèses, sont déterminants.

3.5.5 Contrainte d'observabilité CO

La contrainte CO est, en fait, composée de deux contraintes antagonistes appliquées sur la mémoire, que nous allons spécifier.

1. Une **condition nécessaire**, pour que la sortie du processus de sélection soit une information perceptive, est que les caractéristiques de la mémoire en entrée de ce dernier permettent de garantir que la production d'une sortie S(t) par le mécanisme de sélection [1] est un événement **théoriquement** rare
2. Une autre condition nécessaire est que la non production d'une sortie S(t) soit un événement rare **en pratique**

Il faut expliquer la différence entre les deux termes « théorique » et « pratique ». Dans le premier cas, on considère qu'un signal X quelconque, mis à l'entrée du processus de sélection, n'aboutit que **rarement** à la détection d'une information perceptive. En d'autres termes, l'ensemble des signaux aboutissant à une sortie $S(t)$ est beaucoup plus petit que l'ensemble total des signaux. Il faut rapprocher cette restriction de notre volonté d'éviter, avec une forte probabilité, une fausse détection d'un problème de catégorie 2, dans le problème « D ». Nous assimilons donc le problème consistant à **anticiper l'évolution d'un signal quelconque** à un problème de catégorie 1, ce qui semble réaliste. Au contraire, dans le second cas, on considère qu'un signal X, **expérimenté en réalité** par le système, aboutit presque toujours à la détection d'une information perceptive. Cela signifie que la mémoire du système est construite de manière à ce que le mécanisme de sélection aboutisse, presque sûrement, à une sortie $S(t)$ non vide : cela est en relation avec le fait de détecter, d'une manière fiable, un problème de catégorie 2 dans le problème « D ».

Il faut noter que notre raisonnement est fondé sur l'hypothèse (pragmatique) que l'ensemble des signaux (évoluant dans le temps) auxquels le système fera **réellement** face est bien plus petit que l'ensemble des signaux imaginables.

Nous voyons le rôle central de la mémoire, donc du choix des hypothèses d'évolution : elle doit être construite de manière à respecter la double exigence suivante :

– l'ensemble des hypothèses ne doit pas être trop gros par rapport à l'ensemble total des

1. Voir le chapitre précédent pour la description du mécanisme de sélection. Le terme S(t) est une notation issue également de ce chapitre.

hypothèses imaginables (**contrainte 1**)
- l'ensemble des hypothèses doit permettre de dégager une information perceptive pour tout signal qui est mis réellement en entrée du système (**contrainte 2**)

Dans notre modélisation, lorsque h est fixé, l'ensemble des signaux imaginables, qui ont tous des valeurs dans [0,1] par hypothèse, est isomorphe à $[0,1]^h$ [1]. La contrainte 1 s'exprime en calculant le volume de l'ensemble des vecteurs $(x_{t-h+1},x_{t-h+2},...,x_t)$ mis en entrée du processus de catégorisation, produisant un ensemble S(t) non vide. La contrainte de rareté s'applique sur la mémoire de manière à ce que ce volume soit très petit par rapport au volume de l'ensemble des signaux imaginables. Il faut noter que cette contrainte est indépendante de la nature des signaux en entrée du système : elle est inhérente à la mémoire.

La deuxième contrainte est liée à une bonne adéquation entre la mémoire et les vecteurs $(x_{t-h+1},x_{t-h+2},...,x_t)$ qui vont être utilisés réellement par le processus de catégorisation. Celle-ci tient compte de l'interaction, pendant l'expérience, entre le système et son environnement [2].

3.5.6 Contrainte d'unicité (CU)

CO donne une première contrainte permettant de qualifier la validité d'une mémoire : cette contrainte restreint l'ensemble des paramètres h, i et l admissibles pour chacune des hypothèses. Dans le cas où la mémoire est composée de plusieurs hypothèses, il est possible que l'algorithme de sélection valide plus d'une hypothèse. Du point de vue de la contrainte CO, celles-ci sont toutes identiquement valides : il n'est pas possible de les discriminer. À priori, l'information perceptive, telle qu'elle est limitée par CO, est l'ensemble des hypothèses **validées simultanément**. Nous souhaitons contraindre davantage la notion d'information perceptive. L'idée est de garantir que si le processus de catégorisation détecte une information perceptive, alors il est peu probable que la nature de celle-ci puise être mise en cause (le système ne se trompe pas de catégorie).

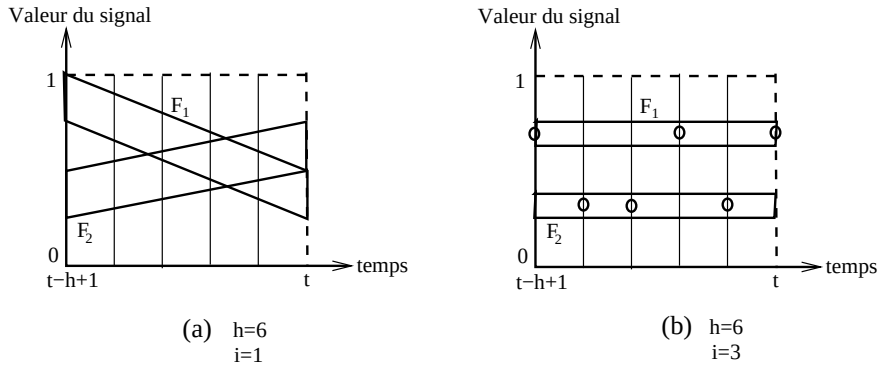
La contrainte d'unicité s'exprime ainsi :

> *Un ensemble d'informations perceptives est valide si aucun événement élémentaire n'appartient à la fois à deux éléments ou plus de cet ensemble.*

La figure 3.6 donne une première idée de la contrainte d'unicité. Le graphe (a) montre deux hypothèses ne pouvant jamais être validées simultanément : la contrainte d'unicité autorisera l'existence de deux informations perceptives à partir de ces deux hypothèses. Par contre, le graphe (b) montre deux hypothèses validables simultanément : la contrainte d'unicité n'autorisera l'existence que d'une information perceptive, au plus [3].

Voici les conséquences de CU. Deux hypothèses x et y ne peuvent être validées simultanément que si elles sont les éléments d'une même information perceptive. Par conséquent, si on considère les ensembles E_x et E_y des signaux aboutissant respectivement à la détection de x et de y, la condition CU impose deux cas :
- E_x et E_y sont disjoints
- E_x et E_y sont égaux

1. Il s'agit de l'ensemble des vecteurs $(x_{t-h+1},x_{t-h+2},...,x_t)$ possibles. 2. C'est-à-dire des signaux qui sont mis réellement en entrée du processus de catégorisation. 3. Nous disons « au plus », car il se peut que la deuxième contrainte afférente à CO ne soit pas satisfaite.



Les focus sont représentés par des parallélogrammes. Les cercles, dans le graphe (b), désignent les valeurs du signal récupérées. Le graphe (a) montre un cas où deux hypothèses, associées respectivement au focus F_1 et au focus F_2 et aux paramètres h=6 et i=1, ne peuvent pas être validées simultanément à l'instant t. Par contre, le graphe (b) montre qu'il est possible de valider simultanément deux focus, pourtant totalement disjoints, lorsque h=6 et i=3. Dans le premier cas, la contrainte d'unicité est respectée, quel que soit l'ensemble des signaux reçus réellement par le système. Dans l'autre cas, la contrainte d'unicité imposera qu'il n'existe qu'une seule information perceptive, regroupant les deux hypothèses.

FIG. 3.6 – Deux exemples d'application de la contrainte CU.

En effet, admettons qu'il existe deux signaux X_1 et X_2 de E_x tels que X_1 valide x et ne valide pas y, alors que X_2 valide x et y simultanément ($X_2 \in E_y$). Cela signifie que l'information perceptive obtenue à partir de X_1 possède comme élément x, mais ne possède pas y. Alors que l'information perceptive issue de X_2 possède x et y comme éléments. Cela signifie que x appartient à deux informations perceptives différentes, ce que CU défend.

La contrainte CU effectue donc une **catégorisation des signaux**, imposant des **catégories disjointes**. Cependant, il faut noter que chacune des catégories pourra comporter plus d'une hypothèse. D'autre part, elle tend à limiter le nombre de catégories possibles. On sent intuitivement (et il reste à le prouver) que l'ensemble CO + CU va impliquer l'existence d'un nombre fini d'informations perceptives, donc de catégories.

3.6 Caractère générique de notre modélisation - Lien avec des modélisations paramétriques existantes

La section 3.2 présente une modélisation du processus de catégorisation. Elle utilise des concepts (modèle à base d'hypothèses, prédiction, durée de validation d'une hypothèse) qui sont à la base de techniques existantes de traitement du signal. Nous souhaitons montrer que notre modélisation du processus de catégorisation possède un caractère de généralité suffisant pour pouvoir traiter un nombre important de problèmes applicatifs, dans la mesure où on maîtrise le moyen de générer une mémoire adéquate[1]. Il convient donc de positionner notre modélisation dans son contexte scientifique : c'est l'objet de cette section.

1. C'est précisément l'enjeu de notre recherche à long terme.

3.6.1 Prédiction : liens avec le filtrage de Kalman

Considérons la méthode du filtrage de Kalman [Kalman, 1960]. Elle est prédictive, donc possède en elle-même la notion d'anticipation qui est à la base de notre modélisation. D'autre part, elle prend en compte l'incertitude liée à l'imprécision relative des données et de la prédiction. Mais les hypothèses sont fortes : un modèle (linéaire pour le filtre de Kalman de base ou linéarisé localement pour le filtre de Kalman étendu) du système doit être disponible. D'autre part, les imprécisions sur le modèle ainsi que sur les mesures doivent pouvoir être assimilées à des bruits blancs gaussiens, pour que le caractère d'optimalité soit assuré. Si ces conditions sont respectées, l'augmentation du nombre de mesures (prise en compte du passé) permet de gagner en précision sur la détermination de l'état courant ainsi que sur la prédiction de l'état futur du système. Nous voyons ici que le nombre de mesures est lié avec la précision du résultat (c'est-à-dire l'écart par rapport à la solution théorique).

Dans notre cas, nous avons aussi besoin d'un certain nombre de valeurs. Cependant, il faut noter que ces valeurs sont utilisées d'une manière radicalement différentes. Ainsi, dans notre démarche, l'information contenue dans la valeur du signal à un instant t n'est pas utilisée (dans le cadre d'un calcul) : chacune des hypothèses détecte la présence de cette valeur dans le focus qui lui est associé. C'est là l'unique utilisation de la valeur du signal. L'accumulation d'informations ne permet pas de gagner en précision (en termes de distance par rapport à un optimal), mais en fiabilité (en termes de probabilité de validation ou d'invalidation d'une hypothèse). Nous ne souhaitons pas nous approcher le plus près possible d'un optimal, mais parvenir à une quasi-certitude sur la coïncidence du signal avec une trajectoire pré-établie (le focus) associée à une hypothèse sur une période de temps h donnée, avec une imprécision connue à l'avance (largeur l du focus).

D'autre part, le filtrage de Kalman sous-entend que l'état du système est unique à chaque pas de temps (même s'il ne peut être appréhendé qu'en termes de probabilités [Jacobs, 1993]). Cette unicité est déduite directement de l'unicité de la perception à un instant t (le système capte une valeur et une seule, pour un signal mono-dimensionnel). Dans notre cas, nous supposons que l'état du système (déduit de l'information perceptive) ne résulte pas obligatoirement de la détection d'une unique hypothèse. En effet, plusieurs d'entre-elles peuvent être sélectionnées simultanément. D'autre part, les hypothèses ne sont pas forcément associées à une durée de validation h unique : le signal est perçu suivant plusieurs résolutions, qui sont toutes aussi valables les unes que les autres (toutes les hypothèses validées sont associées au même degré de fiabilité, et non pas au même degré de précision !).

Enfin, la non prise en compte de la précision du résultat implique que nous ne soyons pas obligés d'émettre des hypothèses restrictives sur la nature du bruit [1]. En outre, nous remarquons que les bruits induits, d'une part par la non-correspondance exacte du modèle à la réalité, et d'autre part par l'imprécision des mesures forment deux entités distinctes dans le filtrage de Kalman. Ainsi, il y a dans l'esprit de cette méthode l'idée qu'on peut distinguer le modèle de la dynamique du système du modèle des mesures issues des capteurs : cela suppose qu'il existe en théorie un modèle parfait et des mesures parfaites. Dans notre cas, nous avons spécifié qu'on ne peut pas distinguer les deux imprécisions et qu'elles n'en forment qu'une seule : la signification de l'invalidation d'une hypothèse signifie simplement que le modèle d'évolution du signal sur une

1. Notre démarche nous oblige à fixer des contraintes sur le système lui-même et non sur son environnement.

période de temps déterminée induit par celle-ci ne correspond pas à la réalité, avec un degré de fiabilité fixé *a priori*. Cette non-correspondance peut être due soit à l'imprécision des mesures, soit à l'imprécision du modèle sur cette plage de temps.

3.6.2 Possibilité de bifurcation

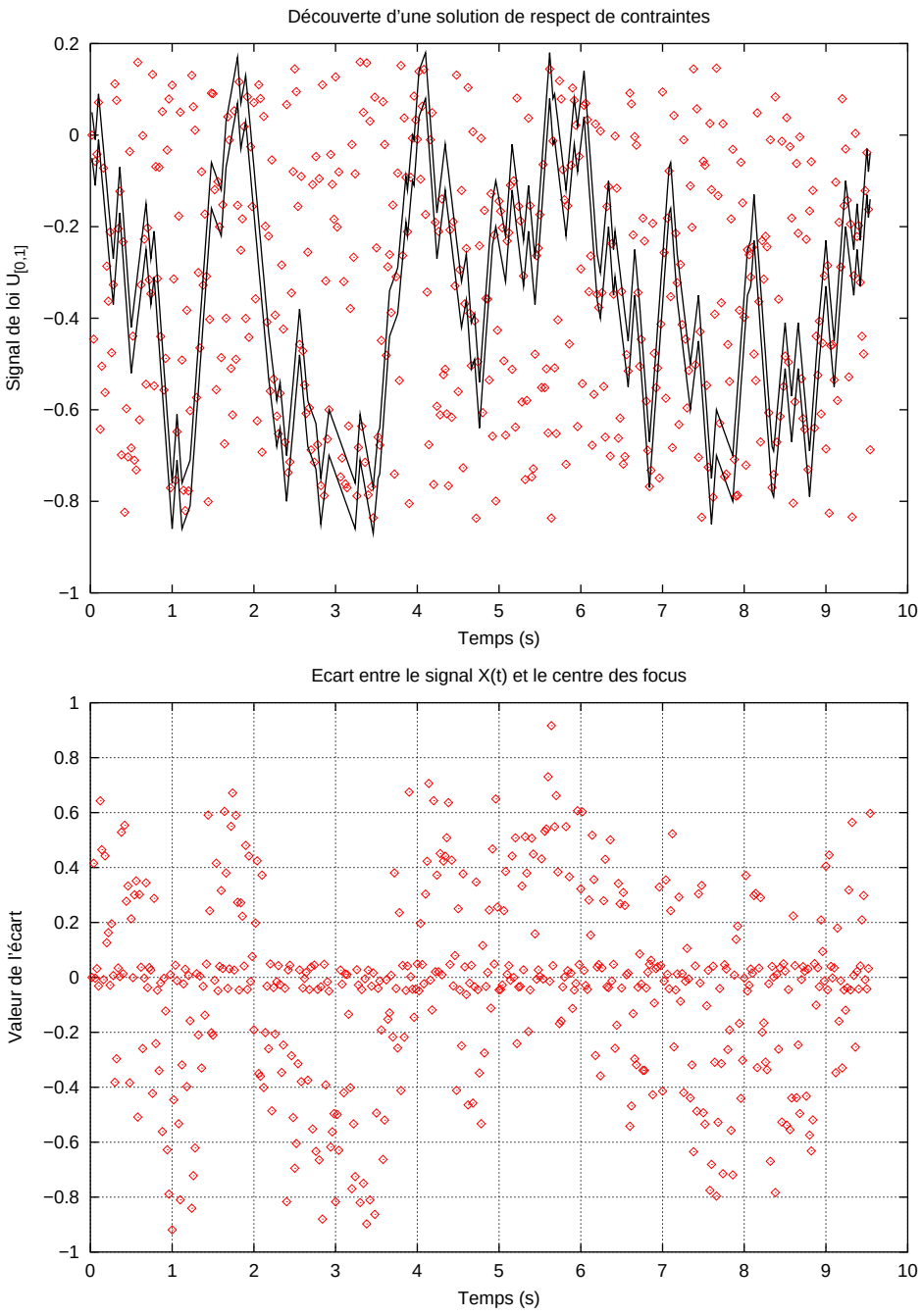
Le sous-système d'apprentissage perceptif montre des caractéristiques de prédiction, de par sa construction. À la limite, si les capacités de prédiction étaient parfaites (condition laplacienne), une hypothèse suffirait (la génératrice du focus serait alors centrée sur la « trajectoire » du signal). Cela signifierait qu'on sait modéliser l'évolution du signal dans le temps, sur des durées importantes. Or, bien évidemment, cela est irréaliste : il est raisonnable de penser que des capacités de prédiction existent sur de courtes durées (donc sur un nombre relativement faible de pas de temps consécutifs), mais pas sur une longue durée.

Que signifie concrètement l'existence de plusieurs hypothèses ? Cela signifie qu'on autorise le système à ne pas savoir prédire, à un moment donné, **une unique** évolution du signal. Le système d'hypothèses donne le choix entre plusieurs évolutions possibles. Plus le nombre d'hypothèses de la mémoire est important et plus on « aide » les capacités de prédiction du système, en lui offrant un choix d'hypothèses qui sera validé *a posteriori*. Pour donner une image de l'indécision du système quant à la prédiction de l'évolution future du signal, on peut considérer que le système est le passager d'une voiture (la dynamique du signal), dont le chauffeur emprunte une route unique (la prédiction sur la direction de la voiture dans un futur proche n'utilise qu'une hypothèse), jusqu'à la survenue d'une bifurcation[1]. Au moment où la bifurcation apparaît, le passager ne sait pas *a priori* quelle direction le chauffeur va emprunter. Par contre, il peut anticiper le fait que le chauffeur va emprunter l'une des voies possibles. Il connaît donc l'ensemble des situations qui vont effectivement se produire : donc, dans un sens plus restreint, il anticipe la situation future en limitant le nombre de cas possibles, sans pouvoir réduire cet ensemble à une unique possibilité.

On se rend facilement compte qu'un ajout trop important d'hypothèses anéantit la faculté de prédiction : il signifie, à la limite, que le système n'a besoin de rien savoir pour parvenir à valider une de ses hypothèses. Ce phénomène est illustré par la figure 3.7. On y représente un système ayant une mémoire comportant tellement d'hypothèses que celles-ci forment un arbre de recherche exhaustif sur les instants futurs, permettant de toujours valider au moins une hypothèse. Les contraintes que nous allons spécifier dans le chapitre suivant ont pour objectif de ne pas autoriser la création d'une mémoire comportant un ensemble d'hypothèses trop riche[2].

En résumé, la possibilité d'ajout d'un nombre important d'hypothèses forme **un degré de liberté supplémentaire** pour le processus de catégorisation. Par contre, notre méthodologie imposera une contrainte à celui-ci, pour que les caractéristiques de prédiction, signifiant que le système utilise des connaissances apprises (la mémoire), ne soient pas détruites par l'accumulation d'hypothèses.

1. Ce terme est employé, en relation avec la théorie des bifurcations. 2. Cela n'est pas vraiment lié avec le nombre d'hypothèses. Nous montrerons que, très probablement, des ensembles comportant un nombre infini d'hypothèses peuvent être admissibles et éviter le résultat que montre la figure 3.7.



On injecte à l'entrée du processus de catégorisation des valeurs aléatoires selon une loi uniforme sur [0,1]. Les focus associés aux hypothèses sont, dans ce cas, tellement nombreux, qu'on peut toujours en trouver un dans lequel un nombre important de valeurs du signal se trouvent. C'est ce qu'illustre le graphe du haut (les deux courbes en traits pleins décrivent les focus sélectionnés à chaque instant. Une rupture dans ces courbes est caractéristique du changement d'hypothèse sélectionnée.

Fig. 3.7 – Mise en évidence d'une solution « absurde » lorsque le nombre d'hypothèses est trop élevé.

3.6.3 Possibilité d'avoir des hypothèses possédant des valeurs de h différentes : lien avec les approches multi-résolutions

Nous avons vu que chacune des hypothèses est associée à une valeur de h qui lui est propre. Ainsi, la mémoire peut comprendre des hypothèses possédant des valeurs de h différentes les unes des autres. Cela signifie que l'information perceptive peut découler d'un ensemble d'hypothèses sur l'évolution du signal à des échelles de temps différentes. La notion d'étude simultanée d'un signal, vu à plusieurs « hauteurs », est à la base de la théorie des ondelettes[1]. Dans ce cadre, un signal est vu comme la combinaison d'un motif de base à des échelles temporelles (fréquences) différentes. L'idée permettant, par exemple, d'effectuer une compression de données à partir d'un signal est de considérer que seules certaines fréquences sont représentatives. Notre motif correspond à la trajectoire associée à une hypothèse. Les fréquences sont associées aux différentes valeurs de h. Dans notre cas, la mémoire contient *a priori* l'ensemble des fréquences intéressantes, puis la sélection affine ce choix initial.

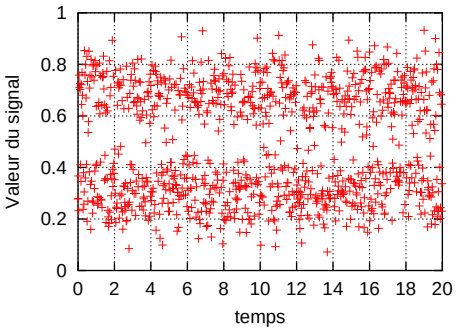
Dans la transformation d'un signal en ondelettes, on est capable de caractériser l'importance de certaines fréquences en termes de conservation relative de la précision du signal obtenu en excluant les termes associés à la majorité des fréquences. Comme dans le paragraphe précédent, c'est la « mesure » qui décide et non pas un ensemble de contraintes. Dans notre cas, il se peut parfaitement que les hypothèses validées ne permettent pas de reconstituer fidèlement le signal d'origine. Cela est dû au fait que l'obtention d'une information perceptive n'est pas liée à l'exigence d'une reconstruction fidèle *a posteriori* du signal.

3.6.4 Possibilité de valider simultanément plusieurs hypothèses : lien avec la problématique de la séparation de sources

La séparation de sources[2] consiste à isoler, à partir des données provenant de plusieurs capteurs, des sources d'information non liées à chacun de ces capteurs. Une image simple de ce processus est l'aptitude d'une personne à **focaliser** son attention sur les propos d'un individu avec lequel on discute, en éliminant toutes informations « perturbatrices » provenant d'autres personnes qui parlent autour d'elles. On peut ainsi isoler un bruit, ou une composante des signaux, qui n'est pas utile.

Dans notre cas, un unique signal en entrée du système peut valider plusieurs hypothèses, c'est-à-dire plusieurs évolutions possibles. La figure 3.8 donne un exemple simple de signal à partir duquel on pourra extraire plusieurs types d'informations perceptives différentes : il s'agit d'un signal mono-dimensionnel, formé de deux sources de distribution gaussienne. Selon la mémoire du système, on pourra, au choix : isoler la source de centre 0.7, isoler la source de centre 0.3 ou considérer les deux composantes. Ainsi, on peut imaginer deux catégories d'hypothèses (dont les focus auront une génératrice constante et centrée soit sur 0.7, soit sur 0.3) qui permettront d'aboutir à la détection d'une information perceptive correspondant à chacun de ces cas.

1. Pour une introduction sur la théorie des ondelettes, le lecteur pourra consulter [Daubechies, 1992]. 2. Le lecteur pourra se reporter à [Jutten et Herault, 1991] ou [Pham et al., 1992].



Les deux sources ont une distribution gaussienne d'amplitude $\sigma = 0.08$ et leurs centres sont constants dans le temps et valent respectivement 0.7 et 0.3 .

Fig. 3.8 – Signal composé à partir d'une distribution gaussienne bimodale.

3.7 Conclusion

3.7.1 Résultats obtenus

Nous avons établi un modèle paramétrique du processus de catégorisation. Il est centré sur un mécanisme de sélection dont le rôle est de valider ou d'invalider un ensemble d'hypothèses d'évolution du signal formant la mémoire du système. Chaque hypothèse est spécifiée à partir d'un triplet (h,i,l) et d'un focus : la valeur de h est la durée de la prédiction, le focus est un cylindre dont la génératrice C(t) est une fonction à valeurs dans [0,1] et dont la section est le paramètre l. L'hypothèse est validée au bout de h pas de temps si au moins h-i valeurs du signal appartiennent au focus associé à celle-ci.

Nous avons présenté deux algorithmes : l'un est applicable lorsque le nombre d'hypothèses contenues dans la mémoire est fini ; l'autre est utilisable lorsqu'il existe un triplet (h,i,l) unique associé à l'ensemble des hypothèses et que les génératrices des focus sont paramétrées par un nombre fini de paramètres à valeurs réelles (cas particulier d'ensembles infinis d'hypothèses). Aucun des deux algorithmes ne possède de paramètres internes [1]

Nous avons ensuite suggéré que notre modélisation pourrait être utilisée dans des domaines applicatifs variés. En effet, elle possède un ensemble de caractéristiques communes avec des modélisations paramétriques existantes. Mais, rappelons-nous que l'objectif du processus de catégorisation est de fournir une information perceptive, associée à des contraintes. Ce chapitre nous montre uniquement comment mettre en oeuvre le processus de catégorisation, mais il ne restreint pas la composition de la mémoire. Nous rappelons ici que l'originalité de notre approche est, à partir d'un modèle paramétrique, de contraindre l'ensemble des paramètres de manière à ce que les solutions issues de ce modèle possèdent, avec une très forte probabilité, un ensemble de propriétés déterminées *a priori*.

1. À l'exception du paramètre δ qui gère la précision de l'approximation de l'ensemble S(t) dans l'algorithme 3.2. Mais la valeur de ce paramètre n'a pas *a priori* une importance capitale : il suffit qu'elle soit suffisamment faible pour approximer correctement S(t).

3.7.2 Travaux à effectuer

L'algorithme 3.2 donne une amélioration certaine par rapport à un simple maillage de l'espace des vecteurs α, en termes de précision de l'ensemble S(t) et de vitesse d'obtention de S(t). Néanmoins, il reste lourd : pour une valeur de h égale à 100 et une valeur de i égale à 30 (ce qui signifie que le système 3.1 possède 100 relations (I_k)), on peut calculer environ une solution S(t) toutes les deux secondes [1]. Cela signifie qu'une information perceptive peut être transmise avec une fréquence maximum de 0.5 Hz. Pour certaines applications, cela peut être clairement insuffisant. Une amélioration possible sera la suivante. Si on constate qu'entre les instants t et t+1, le système 3.1 n'est pas beaucoup modifié [2] (ce qui n'est pas forcément vrai dans le cas général), on peut utiliser le découpage effectué à l'instant t comme base de calcul à l'instant t+1. L'idée est de construire un algorithme donnant itérativement l'ensemble S(t+1) à partir de l'ensemble S(t). Si les différences sont minimes, il est possible que peu de calculs soient à faire entre ces deux instants, ce qui pourrait accroître sensiblement la rapidité de l'algorithme.

L'algorithme 3.2 peut être également amélioré pour inclure des groupes d'hypothèses associés chacun à un triplet (h,i,l). Il n'y a pas *a priori* de changement profond à effectuer sur l'algorithme pour obtenir ce résultat plus général. Le moyen le plus simple est de séparer le calcul de S(t) en autant de calculs associés un triplet (h,i,l) unique. De la même façon, on peut considérer plusieurs familles de génératrices. Ainsi, le calcul de S(t) pourrait être généralisé à une large catégorie de mémoires.

Enfin, il manque un point de travail très important : il s'agit de comparer les performances du processus de catégorisation face à celle des techniques existantes. Ce travail n'a pas de signification pour l'instant, pour les raisons suivantes :

- nous avons autorisé l'existence d'un nombre élevé de degrés de liberté pour le processus de catégorisation. Dans l'absolu, il semble possible de trouver « à la main » des mémoires permettant de résoudre des problèmes particuliers.
- mais, on recherche une méthode opérationnelle pour déterminer la composition de la mémoire, permettant d'éviter un choix arbitraire de la mémoire.

Donc, il aurait été possible de montrer des résultats, même de bons résultats (nous en avons !). Mais, cela aurait simplement signifié qu'on aurait pu, par tâtonnement, trouver des solutions satisfaisantes. Mais, ce « tâtonnement » est beaucoup trop empirique : on n'aurait pas su dire pourquoi cela fonctionne.

Références

Carnap, R. (1950). *Logical Foundations of Probability.* University of Chicago Press.

Daubechies, I. (1992). *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics.* SIAM, Philadelphia.

Jacobs, O. (1993). *Introduction to Control Theory.* Oxford University Press.

Jaulin, L. and Walter, E. (1993). Set inversion via interval analysis for non-linear bounded estimation. *Automatica*, 29(4) :1053–1064.

1. Résultat moyen obtenu avec un PC 350 MHz et 128 Mo RAM, sous Linux. 2. Il résulte de cette faible modification que les découpages formés à l'instant t et à l'instant t+1 peuvent être très semblables.

Jaulin, L., Walter, E., and Didrit, O. (1996). Guaranteed robust nonlinear parameter bounding. In *CESA'96 IMACS Multiconference Symposium on Modelling, Analysis and Simulation*, volume 2, pages 1156–1163, Lille.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transaction of the ASME - Journal of Basic Engineering*, pages 35–45.

Moore, R. (1979). *Methods and Applications of Interval Analysis*. SIAM, Philadelphia.

Pham, D., Garrat, P., and Jutten, C. (1992). Separation of a mixture of independent sources through a maximum likelihood approach. In *Proc. EUSIPCO*, pages 771–774.

Ségal, J. (1998). *Théorie de l'information : sciences, techniques et société de la seconde guerre mondiale à l'aube du XXIe siècle*. Thèse de doctorat, Faculté d'Histoire de l'Université Lyon II.

Shannon, C. (1948). A mathematical theory of communication. *Bell System technical Journal*, 27:379–423,623–656.

4

Résultats théoriques et perspectives à propos de l'information perceptive

4.1 Guide du chapitre

Le chapitre précédent nous a permis de spécifier un modèle du sous-système d'apprentissage perceptif. Nous avons indiqué les caractéristiques de la mémoire (ensemble d'hypothèses, appelées *focus*), ainsi que les paramètres h, i et l attachés à ces focus. Nous avons spécifié les contraintes s'appliquant sur la mémoire, appelées contrainte d'observabilité et contrainte d'unicité. Cela a été mis en oeuvre en utilisant la notion d'événement rare, conceptualisée par le problème « D ».

Dans ce chapitre, nous souhaitons donner des caractéristiques propres à des mémoires respectant les contraintes CO et CU. Comme, dans le cas général, celles-ci engendrent un problème de probabilité complexe, nous nous contenterons de résoudre ce problème sur des cas particuliers de mémoires (§4.2).

Les principaux résultats, dans ces cas particuliers, sont les suivants :

1. pour un signal dont les conditions sur la distribution de probabilité sont très larges, existence d'une mémoire vérifiant les contraintes (CO) et (CU)
2. preuve de l'existence d'une valeur minimum de h strictement supérieure à 1 (équivalent du théorème de Shannon sur l'échantillonnage)

Ensuite, nous donnons en perspective des guides permettant de poursuivre notre travail et d'aboutir à une formalisation de la réaction du système (c'est-à-dire de la mémoire).

4.2 Résolution de CO dans le cas d'une mémoire possédant une hypothèse

4.2.1 Introduction

Nous avons souligné, dans la section précédente, que la formalisation de la contrainte CO aboutit très souvent à un problème ouvert de calcul de probabilité. Il existe toutefois des cas pour lesquels cette formalisation est possible. Le cas d'une mémoire possédant une unique hypothèse en est l'exemple le plus simple. L'objet de cette section est d'en présenter l'étude théorique. Nous allons prouver que, moyennant certaines conditions très faibles sur le signal d'entrée X, on peut toujours trouver une mémoire (caractérisée par le triplet (h,i,l) de l'hypothèse, ainsi que

par le focus qui lui est associé) qui permette d'aboutir à la détection d'une information perceptive. Nous montrerons également l'existence d'une borne inférieure pour h, ce qui nous permet d'avancer un équivalent du théorème de Shannon sur l'échantillonnage.

Dans ce cas précis, la contrainte CU est toujours satisfaite, puisqu'une seule information perceptive peut être détectée. Nous déduirons de la proposition d'existence d'une mémoire que **la détection d'une information perceptive est fiable** dans ce cas de figure.

4.2.2 Notations - Formulation des deux contraintes de CO

L'unique hypothèse est caractérisée par ses trois paramètres : h, i et l. Dans ce cas, il apparaît que la nature de la génératrice du focus n'intervient pas dans les calculs de cette section. Nous supposerons donc que la génératrice est quelconque.

On considère que les h valeurs x_t,x_{t+1},...,x_{t+h-1} du signal X, prises entre les instants t et t+h-1, forment un h-échantillon issu de variables aléatoires réelles X_t,X_{t+1},...,$X(t+h-1)$ à valeurs dans [0,1], qu'on suppose indépendantes deux à deux. Nous appellerons « vecteur solution » tout vecteur (x_1,x_2,...,xh) respectant les deux contraintes de CO. L'hypothèse de rareté sera associée à un réel $\epsilon \in]0,1[$ [1].

L'expression de la contrainte 1 (voir la sous-section 3.5.5, page 107)impose que l'événement suivant soit rare : « Parmi les h valeurs consécutives d'un signal X quelconque, prises aux instants t, t+1, ..., t+h-1, il existe au plus i valeurs de X à l'extérieur du focus associé à l'hypothèse de paramètres h, i et l ».

Nommons P_1 la probabilité que cet événement survienne. L'hypothèse de rareté impose : $P_1 < \epsilon$. Voici comment notre problème de probabilité peut être interprété en terme de calcul de volume de l'ensemble des solutions vérifiant CO. L'ensemble des valeurs possibles du vecteur (x_t,x_{t+1},...,x_{t+h-1}) est égal à $[0,1]^h$, de volume 1. La probabilité que nous souhaitons calculer correspond au rapport entre le volume de l'ensemble des vecteurs solutions (qui est mesurable) et le volume de l'ensemble des vecteurs possibles. La première contrainte de CO impose donc que le volume des vecteurs solutions puisse être inférieur à ϵ.

Dans le cas où i=0, cet ensemble solution est l'hypercube généré par le focus associé à l'hypothèse. Cet hypercube possède des cotés de longueur l. On en déduit que son volume est l^h, ce qui correspond à la probabilité d'occurrence de la détection d'une information perceptive. Dans le cas général, l'ensemble solution est composé d'un ensemble d'hypercubes **disjoints**. Chaque hypercube correspond à l'ensemble des vecteurs solutions présentant k composantes ($k \leq i$) parmi h en dehors de l'hypercube généré par le focus. Pour k fixé, le volume de tous les hypercubes est identique et vaut $(1-l)^k.l^{h-k}$; le nombre de ces hypercubes est combinatoire et vaut C_h^k.

1. Un événement sera dit « rare », relativement à ϵ, si sa probabilité d'occurrence est inférieure à ϵ. Cette valeur est fixée par l'expérimentateur de manière à ce que cet événement, bien que théoriquement réalisable, ne le soit pas en pratique en raison de sa trop faible probabilité d'occurrence. Nous insistons sur le fait que la spécification de ϵ doit tenir compte de la durée de l'expérience : pour une durée infinie, notre définition de la rareté impose une valeur nulle pour ϵ. Le lecteur pourra se reporter à l'annexe B.2, page 147, pour un calcul liant la valeur de ϵ à la durée de l'expérience.

Les hypercubes étant disjoints deux à deux, on en déduit que le volume total de l'ensemble des solutions est donné par l'expression suivante :

$$\sum_{k=0}^{i} C_h^k (1-l)^k . l^{h-k}$$

La première contrainte de CO impose que ce volume soit inférieur à ϵ. Elle implique donc que les triplets (h,i,l), respectant CO vérifient :

$$\sum_{k=0}^{i} C_h^k (1-l)^k . l^{h-k} \leq \epsilon \tag{4.1}$$

Cette relation aurait pu être obtenue sans utiliser les volumes, mais en associant à l'événement élémentaire « la kième composante x_k du h-échantillon est incluse dans le focus associée à l'hypothèse de paramètres h, i et l » une variable aléatoire discrète suivant une loi de Bernoulli. On peut alors considérer la variable aléatoire discrète représentée par la somme de ces h variables aléatoires supposées indépendantes. Cette nouvelle variable aléatoire suit une loi binomiale de paramètres l et h. La probabilité que nous recherchons correspond à la probabilité que cette somme soit supérieure ou égale à h-i.

Pour formuler la deuxième contrainte, nous allons considérer que l'ensemble des signaux perçus réellement peut être modélisé de la manière suivante. On suppose que l'ensemble des vecteurs (x_{t-h+1},x_{t-h+2},...,x_t) sont des réalisations d'un vecteur Y comportant h variables aléatoires réelles, Y étant mis sous la forme suivante :

$$Y = X_d + B$$

X_d est un vecteur, constant dans le temps, appartenant à l'hypercube $[0,1]^h$ et B est un vecteur de variables aléatoires continues B_1,B_2,...,B_h qu'on supposera indépendantes. Les B_j pourront être associées à des lois de probabilité différentes les unes des autres. Une réalisation de B sera un vecteur, noté b_1,b_2,...,b_h. On considère également le vecteur (p_1,p_2,...,p_h)$\in [0,1]^h$, dont chaque composante p_j définit la probabilité pour que x_{t-h+j} appartienne au focus associé à l'hypothèse de la mémoire :

$$p_j = P(Y \in [C(t-h+j) - l/2, C(t-h+j) + l/2])$$

La seconde contrainte est alors exprimable par la relation suivante :

$$\sum_{E \subset H(m)} \left(\prod_{j \in E} p_j \right) \left(\prod_{j \in \overline{E}} (1-p_j) \right) < \epsilon \tag{4.2}$$

Avec $H(m)$ ensemble des parties de $\{0,1,...,h-1\}$ comportant m éléments, et $\overline{E}$ ensembles des éléments de $\{0,1,...,h-1\}$ n'appartenant pas à E.

Cette relation se simplifie dans le cas particulier où les p_j sont tous égaux. On retrouve alors l'expression d'une loi binomiale, similaire à celle de la relation 4.1.

4.2.3 Théorème d'existence d'une mémoire respectant CO et CU

Nous allons montrer que, sous certaines conditions suffisantes, il est possible de construire une mémoire comportant une hypothèse respectant CO. D'autre part, nous allons montrer que, sous certaines autres conditions suffisantes, aucune mémoire possédant une unique hypothèse ne satisfait CO.

On suppose qu'on connaît précisément la valeur du vecteur X_d spécifié dans la sous-section précédente. Dans ce cas, on construit le focus associé à l'hypothèse de la mémoire, de manière à ce que ses valeurs coïncident avec X_d.

Donnons tout d'abord un exemple, illustrant la proposition qui va suivre (voir la figure 4.1). Dans celui-ci, on suppose que X_d est constante dans le temps et vaut 0.5. Nous utiliserons donc un focus dont la génératrice est X_d (toutes les composantes du focus sont égales à 0.5). La représentation du focus à un instant t est donnée par le segment horizontal épais d'ordonnée 1, vu dans les graphes (a) et (c). On suppose que les B_i suivent toutes la même loi, dont la densité de probabilité diffère suivant les graphes (a), (b) et (c). Les graphes (d), (e) et (f) représentent les visualisations d'une observation du signal dans le temps, issues respectivement des densités de probabilités des graphes (a), (b) et (c). Pour le graphe (b), la courbe de densité de probabilité est confondue avec le segment représentant le focus, qui n'est pas représenté en trait épais. Nous nous intéressons à deux aires différentes :

- l'aire A_1 délimitée verticalement par le segment représentant le focus (hachurée sur le graphe (a)), qui représente la probabilité pour qu'une valeur x choisie aléatoirement suivant une loi uniforme sur [0,1] appartienne au focus.
- l'aire A_2 délimitée par les mêmes points P et Q, concernant la fonction de densité de probabilité. Celle-ci représente les p_j (qui sont tous égaux dans ce cas), c'est-à-dire la probabilité pour qu'une valeur x du signal appartienne au focus.

Les trois graphes montrent trois cas :

1. $A_1 < A_2$
2. $A_1 = A_2$
3. $A_1 > A_2$

Nous montrerons que, dans le premier cas, on peut toujours trouver des valeurs de h et i permettant à la fois que le triplet (h,i,l) vérifie CO mais aussi que la détection de l'information perceptive soit fiable : l'exemple typique de loi de probabilité associée aux B_i permettant ce cas est la loi normale. On remarque que, dans le cas d'une loi normale, n'importe quelle valeur de l nous place dans le cas 1. Le deuxième cas est un cas limite : la densité de probabilité est uniforme sur [0,1]. Dans ce cas, on ne peut pas trouver de triplets (h,i,l). Enfin, le troisième cas est typique de problèmes pour lesquels les données récupérées proviennent de plusieurs sources (deux dans notre exemple). Avec le choix de l du graphe (c), on montre qu'aucun triplet (h,i,l) ne convient ; par contre, si on augmente l, on peut (dans notre exemple) retrouver le cas 1, donc trouver des triplets (h,i,l) répondant à nos deux critères (fiabilité de l'information perceptive et respect de CO). Mais, dans ce cas, **on ne peut pas séparer les sources**. Nous montrerons par la suite qu'il est possible de séparer ces sources en utilisant une mémoire à plusieurs hypothèses, qui pourra être interprétée comme un système de séparation de sources.

Revenons à notre problématique. Voici l'énoncé du théorème d'existence :

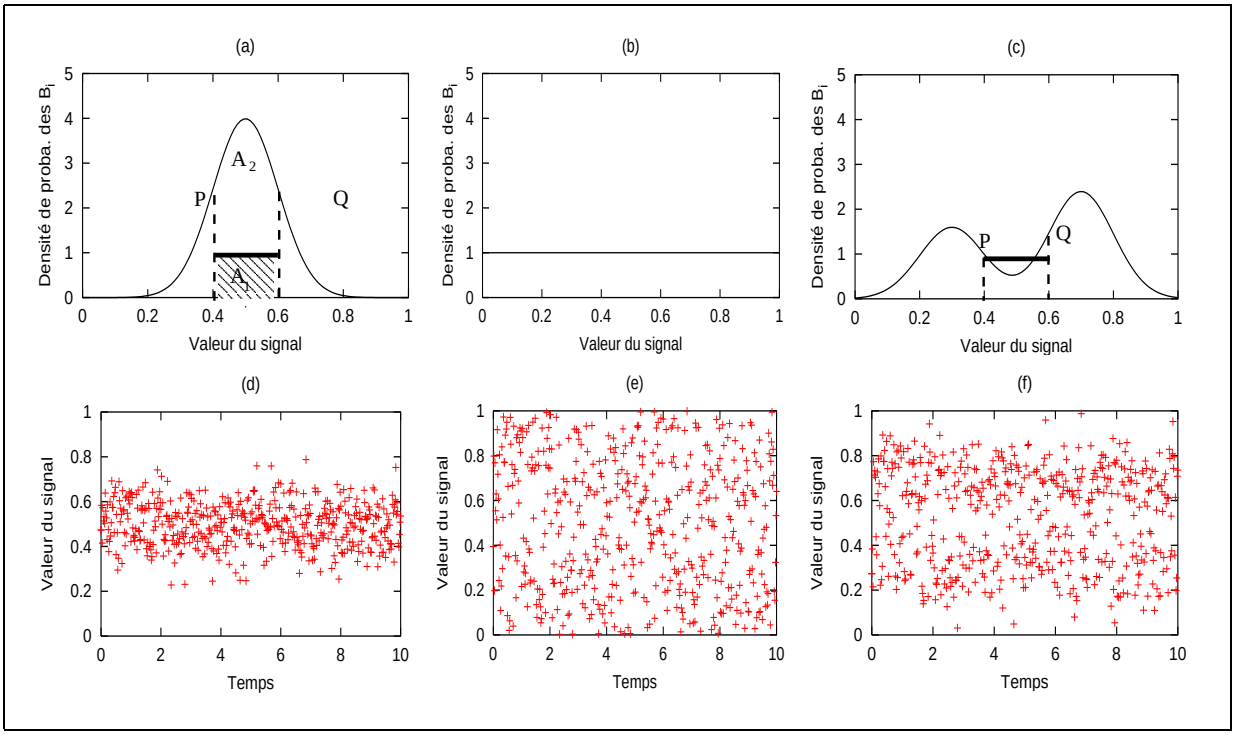


Fig. 4.1 – Trois cas de figure, suivant la densité de probabilité associées aux B_j.

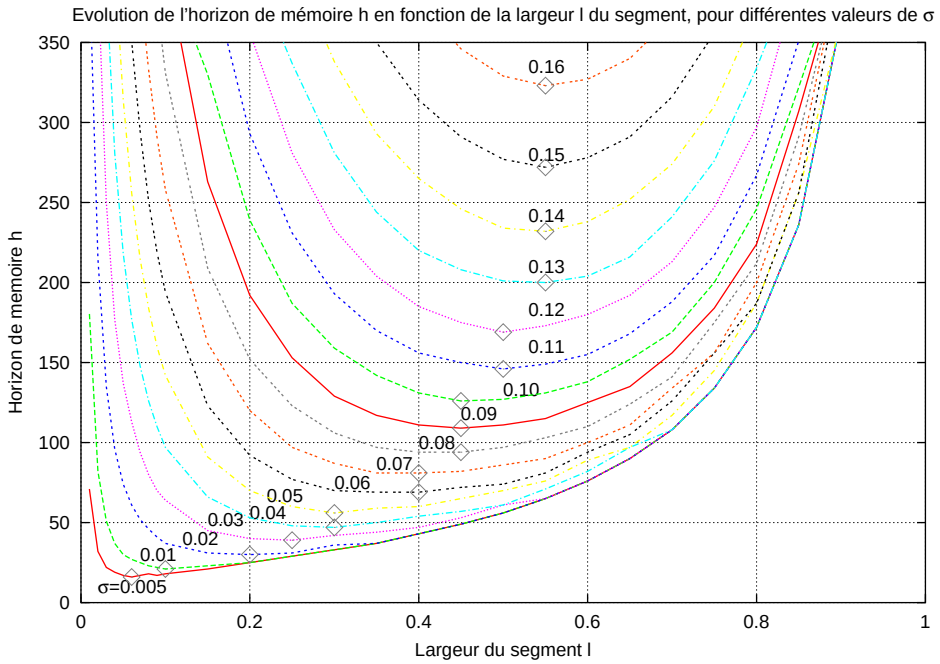
Théorème 1 *Pour tout l dans]0,1[tel que toutes les probabilités p_j sont strictement supérieures à l, alors il existe des valeurs de h et de i telles que (h,i,l) vérifie les deux contraintes associées à CO. D'autre part, pour tout l dans]0,1[tel que toutes les probabilités p_j sont inférieures ou égales à l, il n'existe pas de valeurs pour h et i telles que (h,i,l) vérifie les deux contraintes de CO à la fois.*

Ce théorème signifie que, moyennant certaines conditions suffisantes sur les p_j, on peut trouver une mémoire possédant une unique hypothèse telle que la détection d'une information perceptive à partir du signal X est fiable[1]. La fiabilité de la détection implique que toute observation x_{t-h+1},x_{t-h+2},...,x_t aboutit à la détection d'une information perceptive (celle-ci est unique dans le cas traité par cette section) ; d'autre part, elle implique qu'aucune détection ne survient, en réalité, à partir d'un vecteur quelconque y_1,y_2,...,y_h. Pour compléter cette proposition, on peut également prouver que les triplets (h,i,l) vérifiant CO et ainsi assurant la fiabilité de la détection de l'information perceptive possèdent une valeur de h minimum strictement supérieure à 1. Cette valeur de h dépend de la valeur du paramètre ϵ interprétant la notion de rareté.

Nous préférons ne pas donner la preuve de la proposition 1 dans le corps du document de thèse, car elle est longue et ne présente pas d'intérêt pour la compréhension du cœur de notre travail. Le lecteur pourra se reporter à l'annexe B.3, page 153, pour trouver une démonstration de cette proposition.

La condition suffisante (le cas correspondant à $A_1 < A_2$) permettant d'appliquer le résultat de la proposition 1 est très large. Donnons à présent quelques cas de figure où on peut effectivement

1. Pour cela, il faut bien sûr connaître X_d. Notre proposition prouve l'existence d'une mémoire, sans montrer comment on peut construire la génératrice du focus.



Pour cet exemple, la notion de rareté est associée à $\epsilon = 10^{-15}$. Les croix indiquent la position du minimum des courbes, relatives à une valeur particulière de σ.

Fig. 4.2 – Courbes reliant les paramètres h et l permettant la détection fiable d'un signal de bruit gaussien d'amplitude σ

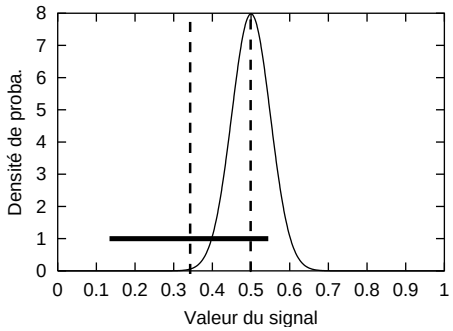
obtenir une détection fiable d'information perceptive :

- le signal X est soumis à un bruit gaussien d'amplitude σ (exemple du graphe (a) de la figure 4.1, page 121). La figure 4.2 montre la relation qui existe entre les paramètre h et l des triplets (h,i,l) valides, lorsque σ varie.
- le signal X est soumis a un bruit uniforme d'une amplitude strictement inférieure à 1
- certaines valeurs x_t du signal X sont aberrantes [1] (avec un pourcentage de données aberrantes strictement inférieur à 1)

Une détection fiable peut donc être mise en place à partir de signaux de mauvaise qualité, possédant même un taux important de données aberrantes. Le prix de cette fiabilité est une augmentation de h et/ou de l. La contrainte CU imposera alors une diminution du nombre d'informations perceptives possibles. Dans le cas où il existe une unique hypothèse, il n'existe donc qu'une information perceptive : la contrainte CU n'intervient pas.

Admettons qu'on construise la mémoire du système [2] à partir d'un modèle pour lequel X_d ne correspond pas tout à fait à la réalité. La fonction génératrice du focus est donc différente de la valeur de X_d réel. Cela introduit un biais dans les B_j réels, qui n'ont plus une espérance nulle. Lorsque l est fixé, l'augmentation du biais aura tendance à diminuer la valeur des p_j. Si ce biais est trop élevé (les p_j sont tous inférieurs à l), on ne pourra plus garantir la fiabilité de l'information perceptive.

1. Dans notre cas, nous considérons qu'une mesure aberrante est obtenue en choisissant au hasard sa valeur avec une loi uniforme sur [0,1]. 2. C'est-à-dire qu'on détermine la génératrice du focus à partir d'un X_d théorique et qu'on calcule h,i et l à partir des B_j de manière à obtenir une information perceptive fiable.



L'écart entre X_d théorique (égal à 0.5) et X_d réel est matérialisé par la distance entre les deux lignes en pointillés.

FIG. 4.3 – Imprécision sur X_d.

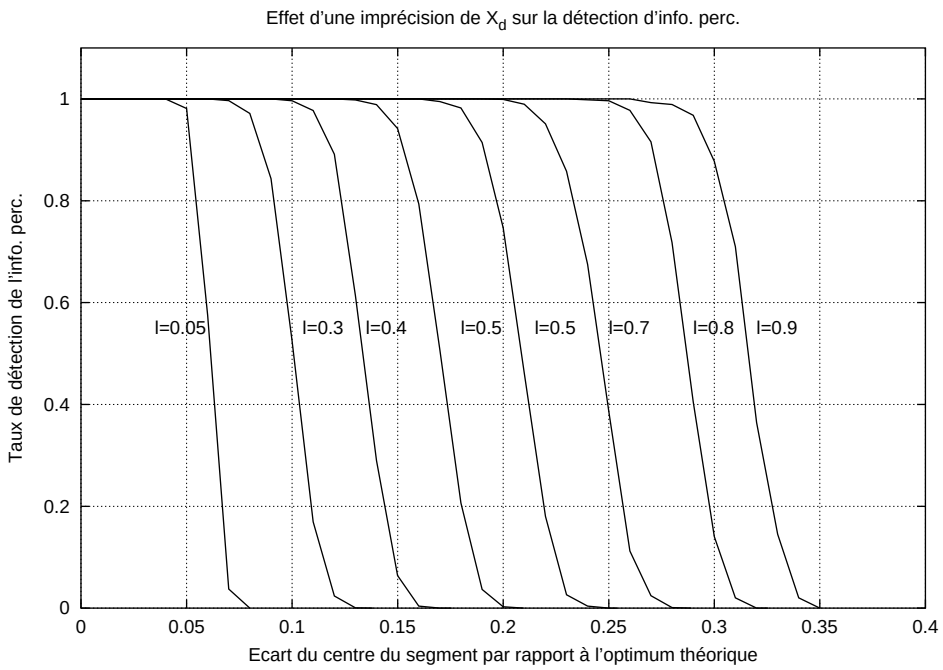
Pour illustrer cela, considérons un ensemble de signaux X dont on sait qu'ils peuvent être modélisés par un X_d constant et des B_j suivant tous une même loi normale de paramètre σ. Dans un premier temps, on suppose qu'on possède une valeur approchée de X_d et la valeur exacte de σ (figure 4.3). On peut déterminer, à partir de σ, un triplet (h,i,l) permettant d'obtenir une information perceptive fiable (dans le cas où X_d est connu exactement). Nous allons étudier l'influence de l'imprécision de X_d sur le taux de détection d'une information perceptive.

Pour cela, lorsque X_d réel est fixé, nous soumettons le processus de sélection au signal X, avec X_d réel $= 0.5$ et $\sigma = 0.05$, pendant 10.000 pas de temps et nous comptons le nombre d'informations perceptives reçues[1]. La figure 4.4 montre les résultats. On constate que le taux de détection de l'information perceptive est maximum jusqu'à une certaine valeur de l'écart entre X_d réel et X_d estimé. Au delà de ce seuil, la performance chute brusquement. La valeur de ce seuil dépend du paramètre l. Lorsque l'écart grandit beaucoup, le taux de détection tend vers 0. Ce résultat n'est pas surprenant. L'information la plus intéressante est la nature des courbes obtenues : elles sont formées de deux plateaux, l'un contenant des valeurs de l'écart associées à un taux de détection maximal, l'autre contenant des valeurs de l'écart associées à un taux de détection nul. La largeur du premier plateau indique l'ensemble des focus associés à la même hypothèse (h,i,l) permettant une détection fiable de l'information perceptive pour le signal X. On peut effectuer une expérience similaire, mais en considérant une imprécision sur la valeur de σ. Les résultats sont donnés par la figure 4.5. Ils sont semblables à ceux trouvés ci-dessus.

4.2.4 Comparaison informelle entre CO et la contrainte imposée par le théorème d'échantillonnage de Shannon

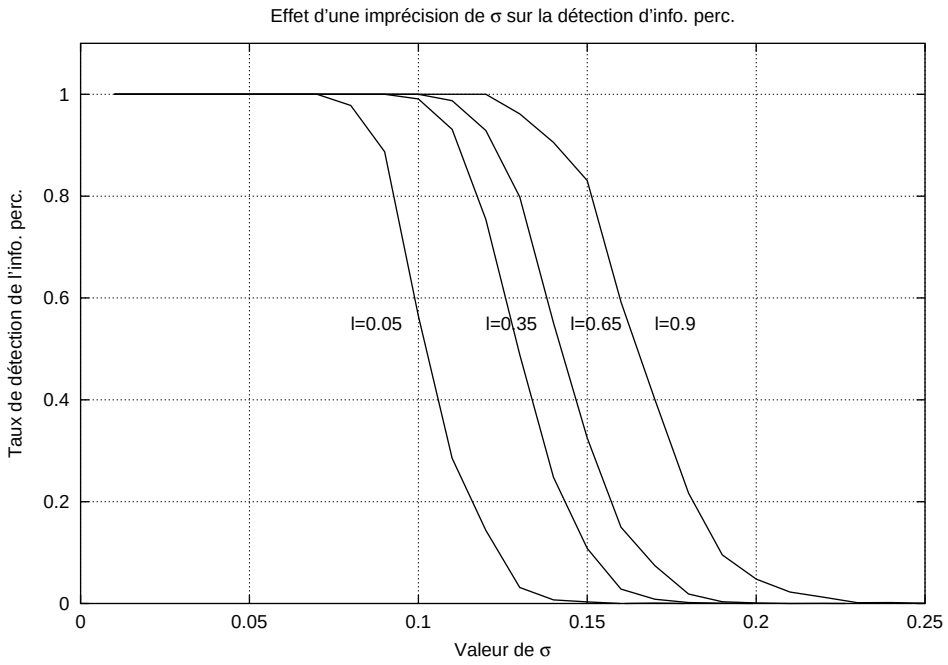
Dans notre exposé, nous avons considéré implicitement que le pas d'échantillonnage est fixé, alors que le paramètre h est variable et signifie « h fois la durée du pas d'échantillonnage ». On peut voir le problème différemment en fixant une durée d de validation de l'hypothèse et en considérant que le pas d'échantillonnage est un paramètre de notre problème. Que cela signifie-t-il ?

1. Pour effectuer cette expérience, nous utilisons l'algorithme de sélection 3.1, indiqué lorsqu'il existe un nombre fini d'hypothèses en mémoire.



La mémoire est construite de manière à permettre la fiabilité de l'information perceptive pour un signal aléatoire gaussien X, de paramètre $\sigma = 0.05$. La notion de fiabilité est associée à une valeur de $\epsilon = 10^{-15}$.

FIG. 4.4 – Évolution du taux de détection de l'information perceptive en fonction de l'imprécision sur X_d



La mémoire est construite de manière à permettre la fiabilité de l'information perceptive pour un signal aléatoire gaussien X, de paramètre $\sigma = 0.05$. La notion de fiabilité est associée à une valeur de $\epsilon = 10^{-15}$.

Fig. 4.5 – Évolution du taux de détection de l'information perceptive en fonction de l'imprécision sur σ

Une hypothèse représente l'évolution supposée d'un signal sur h pas de temps, sans indiquer la durée totale des h pas de temps. Elle définit donc l'évolution du signal à une constante multiplicative de temps près (la valeur du pas de temps). La contrainte CO impose que la valeur de h soit assez élevée relativement à i et l. Par contre, elle ne porte pas sur la durée du pas d'apprentissage. Donc, si on impose une durée d quelconque nécessaire à la validation de l'hypothèse, CO va imposer une valeur de h indépendante de cette durée. Or, nous avons montré, dans le cas d'une mémoire à une hypothèse, que h possède une borne inférieure (pour tout l et pour tout i), dépendante des B_j, donc de la nature de l'imprécision des données autour de X_d (se reporter à la figure 4.2 et à la démonstration de la proposition 1).

Par conséquent, on a montré, dans un cas particulier, que CO permet de donner **une borne supérieure du pas d'échantillonnage, dépendante uniquement de la nature des** B_j. Il y a donc une certaine analogie avec le théorème d'échantillonnage de Shannon. Par contre, il existe trois différences majeures qui font que CO n'intervient pas au même niveau :

- le pas d'échantillonnage ne peut pas être fixé d'une manière absolue, mais relativement à la constante de temps d
- le pas d'échantillonnage ne dépend que des caractéristiques de la mémoire (fixées grâce à CO) et non des caractéristiques du signal reçu par le système
- CO n'est pas une contrainte visant à reconstruire fidèlement le signal reçu à partir de l'information perceptive.

La problématique est donc très différente dans notre cas. Il ne peut pas y avoir la même interprétation physique que dans le cadre du théorème d'échantillonnage de Shannon.

4.2.5 Limitations des mémoires à une hypothèse - Extension du résultat d'existence à une catégorie d'ensembles finis d'hypothèses

Une mémoire a une hypothèse ne permet de détecter qu'une information perceptive. L'état du système peut être caractérisé, à chaque instant t, par la détection ou la non détection de l'information perceptive. La contrainte CO impose que l'hypothèse doit être choisie de manière à ce que la non détection soit un événement rare. Par conséquent, si l'hypothèse est adéquate, le système ne possède, en réalité, qu'un unique état, ce qui est un cas inintéressant en vue de l'interfaçage du sous-système d'AP avec celui d'AO. Il nous faut donc étendre nos résultats. *A priori*, nous recherchons l'ensemble d'hypothèses le plus gros possible, satisfaisant CO et CU.

Mais, cela pose rapidement des problèmes d'ordre théorique. En effet, le problème de calcul de probabilité, induit par les relations 4.1 et 4.2, réside essentiellement dans le fait que les détections des différentes hypothèses ne sont pas indépendantes, en général. Si on considère un raisonnement utilisant le volume des solutions, cela revient à savoir calculer la somme de volumes d'hypercubes non disjoints.

Il existe pourtant un cas où on peut prolonger facilement les calculs réalisés sur une mémoire à une hypothèse : il s'agit du cas où l'ensemble des hypothèses ne peuvent pas être validées deux à deux simultanément : il s'agit d'un cas particulier de respect de la contrainte CU (lorsqu'une hypothèse est validée, on est assuré qu'aucune autre n'est validée en même temps). Dans ce cas, la probabilité de validation d'au moins une hypothèse est égale à la somme des probabilités de validation des hypothèses, prises séparément. On retrouve alors des relations analogues à 4.1 et 4.2, à partir desquelles on peut déduire une propriété d'existence analogue à 1.

4.2.6 Conclusion

Nous avons montré qu'une mémoire possédant une unique hypothèse peut théoriquement permettre de détecter d'une manière fiable une information perceptive à partir d'une gamme importante de modèles de signaux. Nous avons donné des conditions suffisantes sur la nature du signal pour garantir la fiabilité de la détection. Des signaux possédant un taux de valeurs aberrantes, même important, peuvent être détectés d'une manière fiable. Cependant, le prix à payer pour cette fiabilité concerne la précision d'une éventuelle reconstruction du signal à partir de l'information perceptive. En effet, nous avons montré sur un exemple simple que, pour un signal donné, l'ensemble des hypothèses permettant la détection de celui-ci peut être important. Les génératrices associées à ces hypothèses donnent chacune une évolution possible du signal, sans qu'on puisse indiquer laquelle est la meilleure. L'ensemble des résultats donnés pour une mémoire à une hypothèse est transposable à une mémoire possédant un ensemble fini d'hypothèses telles que celles-ci ne peuvent jamais être validées deux à deux simultanément.

Quelles sont les conséquences de cette étude sur l'utilisation du sous-système d'AP dans le cadre de l'AO ? Comme l'information perceptive induit l'état du système, la fiabilité de sa détection correspond à la fiabilité de la détection d'un état du système. Ainsi, si la mémoire est correctement formée, nous venons de montrer que l'état du système, transmis à l'AO, est une donnée fiable, même si les signaux d'entrée possèdent un bruit de mesure important et non obligatoirement gaussien. Les expériences présentées dans cette section laissent également penser qu'il existe un nombre important d'hypothèses différentes permettant de répondre à la contrainte CO : on en déduit que l'état du système peut être défini de plusieurs manières différentes. Par contre, nous ne sommes pas encore en mesure actuellement, ni de contraindre la mémoire, ni d'assurer que les états formés à partir des hypothèses en mémoire permettent de construire un graphe d'états respectant la propriété (P_ϵ) pour un problème d'AO précis : c'est l'objet de l'AP, que nous n'avons pas achevé.

4.3 Conclusion générale des deux premières parties de notre document

4.3.1 Méthodologie

L'unité de notre travail réside dans l'étude de la dynamique de systèmes décrits par des modèles paramétriques soumis à des contraintes internes. L'idée est de contraindre le système de manière à ce que les résultats qu'il génère suivent certaines propriétés connues *a priori*, **quelle que soit la nature de son interaction avec son environnement**. Cette démarche révèle tout son intérêt lorsque l'expérimentateur ne maîtrise pas cette interaction, ce qui est le cas, par exemple, des applications mettant en oeuvre des robots mobiles autonomes dans des environnements complexes. L'objectif suivant est de montrer que, lorsque l'interaction avec l'environnement tend à rompre ces contraintes, le système peut toujours réagir pour rétablir le respect de celles-ci. Enfin, la dernière étape du raisonnement consiste à montrer que cette réaction du système peut, si les contraintes sont bien choisies, être interprétée comme un apprentissage. La logique de preuve concernant les résultats de l'apprentissage se fonde sur l'utilisation des propriétés du système inhérentes à ces contraintes, pour montrer comment le système peut évoluer.

4.3.2 Résultats théoriques

Le cadre applicatif de notre méthodologie est l'établissement d'un algorithme d'apprentissage d'actions réflexes ayant des propriétés de fiabilité et de prédictibilité. Nous avons découpé l'apprentissage en deux étapes :

1. l'apprentissage d'objectif (AO), dont la nature est comparable à celle des techniques d'apprentissage par renforcement
2. l'apprentissage perceptif (AP), dont l'objectif est de fournir un contexte d'apprentissage à l'AO, à partir duquel on a montré des propriétés de fiabilité et de prédictibilité de l'algorithme d'AO

Nous avons appliqué notre méthodologie avec succès pour établir un algorithme d'AO, nommé CbL pour *Constraint based Learning*, qui ne possède aucun paramètre interne. Nous avons pu prouver que si la topologie des états du système respecte une certaine contrainte, nommée (P_ϵ), alors CbL répond à nos exigences en termes de prédictibilité et de fiabilité. En outre, dans ce contexte idéal, l'algorithme CbL montre des performances très supérieures à celle de la technique du Q-Learning, qui est un algorithme d'apprentissage par renforcement classique.

L'objectif de l'AP est de fournir, d'une manière fiable, ce contexte idéal à l'AO, ce qui garantirait la fiabilité du système d'apprentissage AP+AO, quelle que soit l'interaction du système avec son environnement. Le processus sécrété par l'AP est appelé *processus de sélection d'informations perceptives*: il s'agit d'une fonctionnalité de bas niveau qui traite un signal d'entrée (provenant de données capteur, par exemple) et qui transmet ce que nous avons dénommé *une information perceptive* à l'AO. Le moteur du sous-système d'AP est un mécanisme de sélection prédictif, qui filtre des hypothèses d'évolution possibles du signal sur une plage de temps donnée. Le filtrage consiste à valider ou invalider, dans le temps, un ensemble d'hypothèses qui constitue ce que nous appelons la *mémoire du système*. Le mécanisme de validation est déterministe et repose sur un paramètre interne à l'hypothèse. La mémoire possède des caractéristiques prédictives, qui peuvent s'appliquer simultanément sur plusieurs échelles de temps. Ainsi, le modèle du sous-système d'AP peut être relié à des techniques existantes du traitement du signal.

Nous avons utilisé notre méthodologie afin de contraindre les paramètres internes de la mémoire, de manière à ce que l'information perceptive détectée soit fiable. Pour cela, nous supposons que celle-ci possède une caractéristique de rareté (cela constitue la contrainte d'observabilité (CO)). Dans le cas de mémoires simples, nous avons montré théoriquement que les paramètres de la mémoire peuvent être effectivement contraints de manière à ce que l'information perceptive soit fiable.

4.3.3 Algorithmes de sélection

Nous avons construit deux algorithmes du mécanisme de sélection. L'un s'applique lorsque la mémoire possède un nombre fini d'éléments et l'autre est utilisé lorsque celle-ci est constitué d'un ensemble infini d'hypothèses, engendrées à l'aide d'un nombre fini de paramètres. Dans ce dernier cas, nous avons montré que la résolution de la contrainte (CO) engendre un ensemble d'hypothèses validées qui est la solution d'un problème d'inversion ensembliste. L'utilisation de l'analyse par intervalles nous permet de trouver deux ensembles d'hypothèses encadrant l'ensemble de solutions au problème inverse : l'algorithme *SIVIA*, de Jaulin et Walter, est utilisé dans cette résolution. Il permet, en outre, de résoudre le problème d'inversion dans des cas où celui-ci ne s'exprime pas à l'aide d'un système linéaire. Il fonctionne par découpages successifs

du problème en boîtes possédant trois états possibles : tous les points de la boîte sont solution du problème inverse, aucun point n'est solution ou il existe un mélange des deux types de points.

Dans les deux cas (fini ou infini), les algorithmes garantissent de trouver l'intégralité des hypothèses validables à chaque pas de temps.

4.4 Perspectives

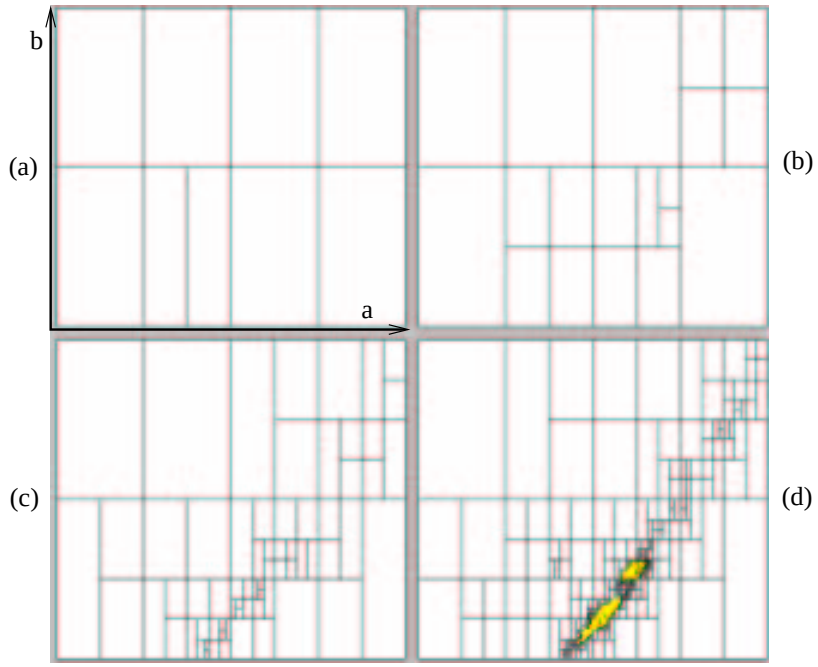
4.4.1 Introduction

Ce document de thèse présente l'avancement actuel d'une recherche qui se place sur une perspective de long terme. Plusieurs difficultés doivent être résolues avant d'aboutir à la réalisation d'un système complet d'apprentissage d'actions réflexes possédant des caractéristiques de fiabilité. Elles concernent l'AP, pour une grande partie d'entres-elles. Cette section fait le point sur les problèmes auxquels nous n'avons pas apporté, pour l'instant, de solution satisfaisante.

4.4.2 Conjecture à propos des mémoires possédant un ensemble infini d'hypothèses

Nous avons constitué un ensemble de contraintes sur la mémoire du système. La principale difficulté que nous rencontrons actuellement est de pouvoir vérifier, pour une mémoire donnée, c'est-à-dire pour un ensemble d'hypothèses fixé, si ces contraintes sont respectées ou non. En effet, notre méthodologie impose que le système réagisse à son interaction avec son environnement, dans le cas où les contraintes qu'il subit ne seraient plus satisfaites : c'est cette réaction que nous souhaitons pouvoir interpréter comme un apprentissage du système.

Dans l'état actuel, nous ne savons pas, dans la plupart des cas, si la contrainte CO est satisfaite ou non. L'expression de cette contrainte est un problème de probabilité qui devient très rapidement ardu à résoudre dans un cadre purement théorique. L'estimation des paramètres pourrait être effectuée d'une manière numérique par une méthode de Monte Carlo : une propriété du système respectant la contrainte CO est que si on injecte en entrée du processus de sélection un vecteur « signal » comportant des valeurs prises aléatoirement suivant une loi uniforme, alors la probabilité pour qu'une information perceptive soit détectée est inférieure à un certaine valeur, fixée à l'avance. Or, pour que notre postulat de rareté soit respecté, il faut que cette valeur soit extrêmement faible, de manière à ne jamais détecter une information perceptive, en pratique, dans ce cas. Nous voyons bien que, pour un ensemble de paramètres donnés de la mémoire, on peut tester son rejet par une méthode de Monte Carlo : il suffit de détecter une information perceptive. Par contre, cette méthode ne pourra jamais, par définition de la rareté, valider un ensemble de paramètres donné. La figure 4.6 présente le résultat du mécanisme de sélection utilisant *SIVIA*, à un instant donné, pour quatre mémoires dont les hypothèses sont générées à partir de deux paramètres (constituant les axes x et y de nos graphes). Pour chacun des graphes, les valeurs de h et l sont identiques. Seul i varie. On s'aperçoit que, dans le dernier cas (graphe (d)), l'algorithme de sélection trouve des solutions (les quatre tâches). Cela suffit pour rejeter le triplet (h,i,l) correspondant à la mémoire utilisée dans ce cas. D'autre part, on remarque que la taille moyenne des rectangles en bleu diminue lorsque i augmente. Dans le cas **d'un système linéaire** (C(t)=a.t+b), il semble donc exister une relation entre la grosseur des pavés pour lesquels il n'y a pas de solution et le nombre d'inéquations satisfaites dans la relation



Les boîtes vides traduisent qu'aucune solution n'existe dans celles-ci ; les zones pleines signifient que certains points peuvent être solutions. Les quatre graphes sont les résultats, au même instant t et pour les mêmes valeurs du signal d'entrée, de quatre mémoires différentes. Les mémoires sont constituées d'un ensemble infini d'hypothèses (les génératrices sont des droites, possédant deux paramètres qu'on fait varier). Les paramètres h et l sont communs et valent respectivement 17 et 0.05. La différence provient de la valeur de i : respectivement 3, 6, 9 et 12. Les données à l'entrée du processus de sélection sont choisies aléatoirement, suivant une loi uniforme sur [0,1].

FIG. 4.6 – Quatre résultats de l'algorithme de sélection

3.1 [1]. Cette piste serait intéressante, car on pourrait ainsi déterminer graphiquement des valeurs approchées de triplets (h,i,l) admissibles.

Peut-on s'attendre à des résultats analogues à ceux de la propriété 1 pour des ensembles infinis d'hypothèses, spécifiés à partir d'un ensemble fini de paramètres ? Peut-on prouver l'existence de triplets (h,i,l) vérifiant CO et CU ? Un premier élément de réponse peut être donné par l'expérience suivante. On fixe *a priori* les valeurs de i et l et on fait varier le paramètre h. Dans le cas où h est assez petit, on peut déterminer, par une méthode de Monte Carlo, la probabilité de détection d'une information perceptive à partir des données choisies aléatoirement selon une loi uniforme sur [0,1] : nous comparons les résultats obtenus pour une mémoire possédant une hypothèse (dont on connaît les propriétés théoriques) avec ceux obtenus pour une mémoire dont les hypothèses sont générées à partir de deux paramètres (mémoire utilisée dans la figure 4.6). La figure 4.7 montre cette comparaison. On s'aperçoit, sans surprise, que, à h,i et l fixés, la probabilité de détection pour une mémoire possédant un ensemble infini d'hypothèses est toujours supérieure à celle obtenue pour une mémoire ayant une unique hypothèse. Le fait intéressant est que le comportement des deux courbes lorsque h augmente est identique (les courbes ont une pente approximativement égale, à h fixé, pour h>10). Cela semble indiquer que les limites

1. Pour la figure 4.6, on remarque nettement que, lorsque i augmente, la taille moyenne des pavés en bleu diminue.

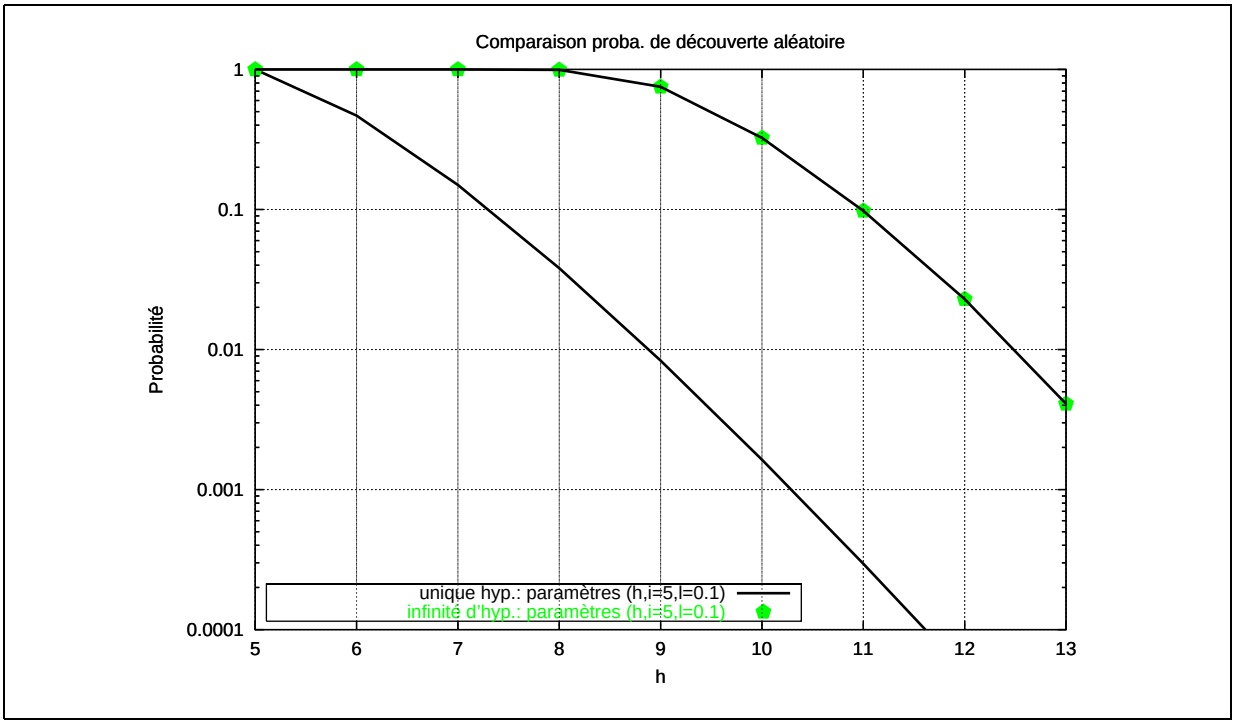


Fig. 4.7 – Évolutions comparées de la probabilité de détection d'information perceptive à partir d'une entrée aléatoire.

asymptotiques des deux probabilités de détection sont proches, donc que le comportement de la probabilité de détection pour une mémoire de taille infinie est identique à celui d'une mémoire possédant une unique hypothèse. À partir de ce résultat, **nous conjecturons qu'il existe probablement des résultats analogues à la propriété 1 pour certaines catégories de mémoires ayant une taille infinie**.

4.4.3 Piste de recherche sur l'AP

Pour obtenir un algorithme d'AP, il nous faut tout d'abord savoir, à tout moment, si les contraintes (P_ϵ), CO et CU sont respectées. Nous avons abordé le problème de CO dans la sous-section précédente, mais il reste à préciser pour CU. Le respect de (P_ϵ) n'est pas difficile à détecter (voir la méthode de calcul de H_1 et de H_2 dans la première partie de ce document). En admettant que le problème de détection soit résolu, il faut ensuite répondre aux questions suivantes, dans l'ordre dans lequel elles sont posées :

- Comment l'environnement agit-il sur la mémoire ?
- Comment la mémoire réagit-elle, si l'action de l'environnement tend à rompre les contraintes, pour les rétablir ?
- Cette réaction est-elle interprétable comme un apprentissage ?
- Peut-on montrer des propriétés liées à cet apprentissage ?

Le thème central de ces questions est **la manière de modifier la mémoire au fil du temps**. Nous pensons à deux options très différentes. La première consiste à poser une mémoire vierge (ne contenant aucune hypothèse) à l'instant t=0 et à augmenter progressivement le nombre d'hypothèses au fil de l'expérience perceptive du système, en sachant que cette augmentation est modérée par les contraintes CO et CU : nous l'appellerons O1. La deuxième, notée O2, consiste,

au contraire, à considérer, à t=0, la mémoire la plus grosse possible respectant CO, mais ne respectant pas théoriquement CU (l'utilisation des mémoires possédant une infinité d'hypothèses peut être envisagée dans ce cadre). Dans ce cadre, l'interaction avec l'environnement peut montrer que CU n'est pas valide (des hypothèses sont associées à plusieurs informations perceptives), ce qui oblige le système à éliminer des hypothèses (on pourrait raisonner dans l'espace des paramètres des génératrices et imaginer l'utilisation du découpage en boîtes de l'algorithme *SIVIA* pour rejeter un ensemble de boîtes, donc un ensemble d'hypothèses).

L'option O1 semble peu réaliste si on assimile notre mémoire à la mémoire biologique, et les hypothèses à un ensemble de neurones. La seconde est plus plaisante, car elle fait penser au mécanisme de sélection neuronal, mis en évidence par Edelman [1]. Nous préférons, pour cette raison, ne considérer que O2. Voici notre réflexion menée à partir de cette option.

L'idée est de constituer une base initiale d'hypothèses la plus riche possible (en terme de valeurs de h et de l différentes), qui respecte l'intégralité des contraintes avant le début de l'expérience. À notre avis, le terme « la plus riche possible » s'interprète par une notion de maximum s'appliquant à la mémoire: l'ajout d'une hypothèse supplémentaire romprait les contraintes. Nous notons que la deuxième contrainte de CO ne se pose pas avant le début de l'expérience, puisque le système n'a pas encore été confronté à des signaux réels. De même, CU ne se pose pas non plus. La seule condition porte donc sur la première contrainte de CO. Lorsque le système va recevoir son premier signal, jamais expérimenté, la première contrainte de CO va impliquer que le système ne va pas détecter d'information perceptive, en toute probabilité. Par contre, la deuxième contrainte de CO va être rompue, puisqu'elle stipule que la détection d'une information perceptive est assurée **en pratique**. Comment faire pour rétablir la deuxième contrainte de CO? Nous savons, par hypothèse de construction, qu'on ne peut pas rajouter une nouvelle hypothèse en mémoire, sous peine de rompre la première contrainte de CO. L'idée est alors de modifier la génératrice d'un ou de plusieurs focus, de manière à ce que le signal provoque la détection d'une information perceptive. Nous souhaiterions que cette modification s'effectue sans changer le caractère « maximal » de la mémoire. Admettons que nous sachions le faire. Quelles hypothèses faut-il modifier pour assurer cette réaction du système? Dans quelle mesure ce changement ne va pas provoquer un « oubli » de signaux déjà perçus? Combien d'hypothèses faut-il modifier? Illustrons nos propos en utilisant un raisonnement sur les volumes. La première contrainte de CO impose que le volume de l'ensemble total des vecteurs solutions est inférieur à un réel ϵ, qui est associé à l'interprétation de la rareté. Une mémoire de taille maximale est le cas limite pour lequel ce volume est égal à ϵ. Le changement de la nature de certains focus doit s'effectuer à volume constant. En d'autres termes, la modification de la génératrice de certains focus doit provoquer le déplacement d'une partie de l'ensemble des vecteurs solutions vers une zone jusque là inoccupée de l'ensemble des vecteurs solutions. Pour le choix du sous-espace à déplacer, on pourrait imaginer prendre celui qui a été le moins activé sur une certaine période de temps. Au bout d'un certain temps, ce mécanisme va permettre de concentrer le sous-espace des vecteurs solutions **théoriques** au niveau du sous-espace des vecteurs solutions **pratiques**. En d'autres termes, la deuxième contraintes de CO tendrait à être respectée, alors que la première le serait toujours.

1. Les neurones possèdent *a priori* une très grande connectivité, qui diminue fortement en ne conservant que les connexions « solides ». Edelman explique ce phénomène grâce au principe de la sélection naturelle, propre à la théorie évolutionniste.

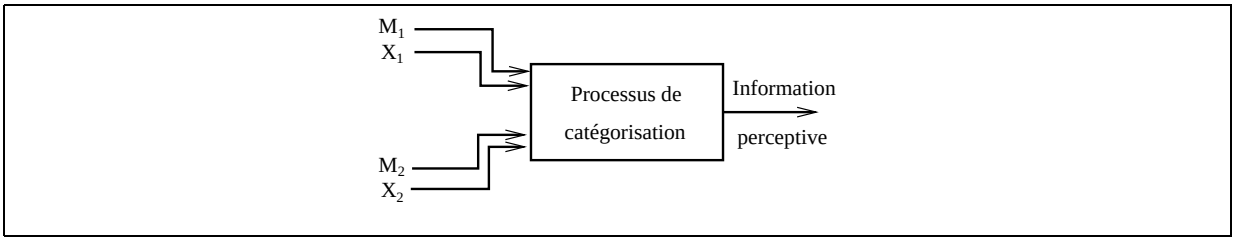


FIG. 4.8 – Processus de sélection possédant deux signaux d'entrée.

Cette réflexion nous semble satisfaisante du point de vue du respect de CO. Mais, elle n'aborde pas CU.

4.4.4 Généralisation à l'utilisation de plusieurs signaux d'entrée

Nous nous sommes limités à l'étude d'un processus de sélection ne possédant qu'un signal d'entrée. Comment faire lorsqu'il existe plusieurs signaux ? Notre idée est de dédier une mémoire particulière à chaque signal (voir la figure 4.8) : nous suivons en cela la spécialisation des différentes zones du cerveau. La mémoire totale est, pour nous, la somme des mémoires prises individuellement. De plus, nous pensons que la première contrainte de CO doit être respectée individuellement, pour chacune des mémoires, alors que la deuxième contrainte de CO et la contrainte CU se comprennent sur l'ensemble des mémoires dédiées chacune à un signal. Voici l'interprétation de cela. Le processus de sélection n'est pas modifié : il valide ou invalide l'ensemble des hypothèses contenues dans les mémoires en entrée du système, relativement aux signaux auxquels celles-ci sont dédiées. Dans notre exemple de la figure 4.8, M_1 sert à valider X_1, alors que M_2 sert à valider X_2. D'autre part, le sous-système d'AP ne fournit qu'une seule information perceptive. La contrainte CU impose qu'une hypothèse ne puisse pas appartenir à deux informations perceptives. Ainsi, si deux hypothèses sont validées simultanément (elles peuvent appartenir à deux mémoires élémentaires différentes), elles sont associées à une unique information perceptive.

Cette analyse est cohérente avec le fait que notre processus de vision nous donne l'impression que la forme d'un objet et sa couleur forment un tout, alors qu'ils sont analysés par des zones différentes du cerveau, mais d'une manière simultanée (pour un objet donné, l'analyse de la forme activerait toujours les mêmes régions du cerveau, alors que l'analyse de la couleur serait effectuée également par une région spécifique).

4.4.5 Genèse de l'information perceptive à l'aide d'actions réflexes

Une idée débattue actuellement dans le domaine des sciences cognitives est le rôle de certaines actions réflexes dans l'obtention d'une information perceptive. L'exemple typique est celui des saccades oculaires, dont on a déterminé l'importance dans les tâches de reconnaissance visuelle (reconnaissance d'un visage, par exemple). Une hypothèse tenable est que ces mouvements seraient discriminants pour l'information perceptive. Voici comment nous intégrons cela à notre modélisation du sous-système d'apprentissage perceptif.

Le fait de bouger le bras, par exemple, est engendré par un signal moteur, mais est accompagné d'une sensation particulière, qui permet de savoir qu'on bouge effectivement le bras. Cette sensation peut être décrite comme un signal qui accompagne le mouvement. Nous souhaitons, tout simplement, placer ce signal à l'entrée du processus de sélection. D'après la sous-section

précédente, il sera accompagné d'une mémoire dédiée. Pourquoi cela change-t-il quelque chose à notre problème? Simplement parce que le système peut maîtriser l'évolution de ce signal dans le temps, sur de longues périodes (le phénomène de bifurcation décrit dans le chapitre précédent, introduisant l'utilité d'avoir un ensemble étoffé d'hypothèses d'évolution, n'apparaît pas ici). Or, l'obtention d'une information perceptive incluant plusieurs hypothèses, dont une qui est issue du signal lié au mouvement, permet d'associer (ou de corréler) les autres hypothèses avec ce mouvement qu'il maîtrise. L'exécution de l'action réflexe joue alors le rôle de bifurcateur, en contrôlant l'évolution des autres signaux d'entrée du processus de sélection. Nous souhaitons appliquer cette modélisation dans le cadre d'une expérience de reconnaissance dynamique d'environnement utilisant un robot mobile Khepera. L'idée générale est que ce robot est incapable, à un instant t donné, de reconnaître avec fiabilité son environnement (il est trop myope). Par contre, nous espérons pouvoir montrer que le robot peut effectuer une reconnaissance fiable, en discriminant les situations perceptives grâce à son mouvement sur un période de temps courte.

B

Éléments relatifs au chapitre 4

B.1 Exemple d'information perceptive pour un signal mono-dimensionnel

B.1.1 Introduction et notations

On considère un signal X mono-dimensionnel dont les valeurs réelles sont bornées. Par convention, on considérera dans la suite de ce chapitre que toutes les valeurs de X sont comprises dans l'intervalle [0,1].
Les valeurs de X sont récupérées à intervalles de temps réguliers, toutes les τ secondes. Les termes $X(t)$,$X(t+1)$,...,$X(t+k)$ désignent respectivement la valeur de X récupérée à l'instant t,$t+\tau$,...,$t+k.\tau$.
Le système perceptif P_X qui réagit au flux de données reçues de X est composé d'un ensemble de n « résolutions » ; $P_X = r_1,r_2,...,r_n$. Pour une résolution r_k donnée, le segment [0,1] est partitionné en r_k segments $i_0,i_1,...,i_{r_k-1}$ de longueur égale à $1/r_k$ (figure B.1) :
pour tout $k \in \{1,...,n\}$,$[0,1] = \bigcup_{j=0}^{r_k-1} i_j = \left(\bigcup_{j=0}^{r_k-2} [\frac{j}{r_k},\frac{j+1}{r_k}[\right) \bigcup [\frac{r_k-1}{r_k},1]$.
Lorsqu'un signal $X(t)$ est reçu par le système perceptif P_X, il touche un et un seul segment i_j pour chaque résolution r_k (figure B.2). Pour chacune de celles-ci, nous allons nous intéresser à la suite des segments touchés successivement, et plus particulièrement à la cohérence de cette suite. Le terme « cohérence » fait ici référence à une prolongation de la notion de continuité dans un cadre où les données sont échantillonnées et bruitées et où, par conséquent, le passage à la limite $\lim_{h\to 0} X(t+h) - X(t)$ n'a plus de sens ($|h|$ est minorée par τ et, les données étant bruitées, la valeur $X(t+h) - X(t)$ est probablement non proche de zéro, même si h est très proche de 0). D'autre part, nous ne souhaitons pas appliquer à X un quelconque modèle de variable aléatoire. Dans notre cas, un signal ne sera perçu que par l'intermédiaire de l'évolution des segments touchés pour chacune des résolutions : la notion de cohérence ne va donc pas concerner l'évolution du signal en lui-même, mais celle des segments touchés.
Pour une résolution r_k donnée, l'évolution du signal X sur un intervalle de temps $[0,t]$ va engendrer une suite de segments touchés $i_{j_1},i_{j_2},...,i_{j_p}$. Cette suite est construite de manière à n'ajouter un segment particulier qu'à l'entrée du signal sur celui-ci (figure B.6) : c'est la transition d'un segment à un autre qui est mise en évidence. Dans le cas où le signal X ne toucherait qu'un seul segment au cours de son évolution, la suite serait réduite à celui-ci.

La notion de métrique, nécessaire pour effectuer un passage à la limite, est ici remplacée par une notion triviale de voisinage

Définition 1 *Deux segments i_k et i_l sont voisins si et seulement si l'intersection de leur adhérence dans [0,1] est réduite à un point.*

En d'autres termes, si $i_k = [a_k,b_k[\subset [0,1]$ et $i_l = [a_l,b_l[\subset [0,1]$, leur adhérence respective est égale aux segments fermés $[a_k,b_k]$ et $[a_l,b_l]$ et l'intersection entre les deux segments ainsi formés est non vide si et seulement si $b_k = a_l$ ou $b_l = a_k$. Cela signifie simplement que les deux segments sont côte-à-côte.
À partir de cette définition, nous exprimons ce que nous entendons par *incohérence*.

Définition 2 *Une suite de segments $i_{j_1},i_{j_2},...,i_{j_p}$ engendrée par l'évolution d'un signal X est dite incohérente si et seulement si il existe $k \in \{1,...,p-1\}$ tel que i_{j_k} et $i_{j_{k+1}}$ ne soient pas voisins.*

Un exemple d'incohérence est donné par la figure B.5. La notion de cohérence découle logiquement de celle d'incohérence

Définition 3 *Une suite de segments $i_{j_1},i_{j_2},...,i_{j_p}$ engendrée par l'évolution d'un signal X est dite cohérente si et seulement si elle n'est pas incohérente suivant la définition 2.*

Il n'est pas difficile de constater que la cohérence dépend de la résolution avec laquelle le signal est perçu. La figure B.7 montre ce fait.
Nous avons défini la manière dont le signal est capté par le système perceptif P_X. Mais, nous n'avons pas encore évoqué la façon dont la dynamique du signal était interprétée. Rappelons que dans la construction d'une suite de segments, chaque élément est ajouté dès qu'une transition entre le dernier segment touché et cet élément est détectée. Toutefois, cette procédure ne permet pas de dégager les tendances d'évolution du signal : il nous manque la connaissance de la transition de sortie de cet élément. Dans le cas où la suite de segments est cohérente, il n'existe que deux possibilités :

1. la transition de sortie pointe vers le segment initiateur de la transition entrante (figure B.3).
2. la transition de sortie pointe vers l'autre segment voisin que le segment initiateur. Dans ce cas, une tendance d'évolution est constatée.

Cela signifie que lorsqu'on considère trois segments i_j,i_k et i_l faisant partie d'une suite cohérente, nous avons deux cas possibles :

1. $i_j = i_l$, équivalent à la première possibilité citée ci-dessus.
2. $i_j \neq i_l$, équivalent à la deuxième possibilité citée ci-dessus.

Dans ce deuxième cas (voir la figure B.4), nous pouvons définir un sens de parcours du segment i_k.

Définition 4 *Dans le cas où les éléments de i_j sont tous inférieurs à ceux de i_l (le signal traverse i_k de gauche à droite), nous dirons que i_k est orienté positivement, ce que nous noterons i_k^+.*
Au contraire, dans le cas où les éléments de i_j sont tous supérieurs à ceux de i_l (le signal traverse i_k de droite à gauche), nous dirons que i_k est orienté négativement, ce que nous noterons i_k^-.

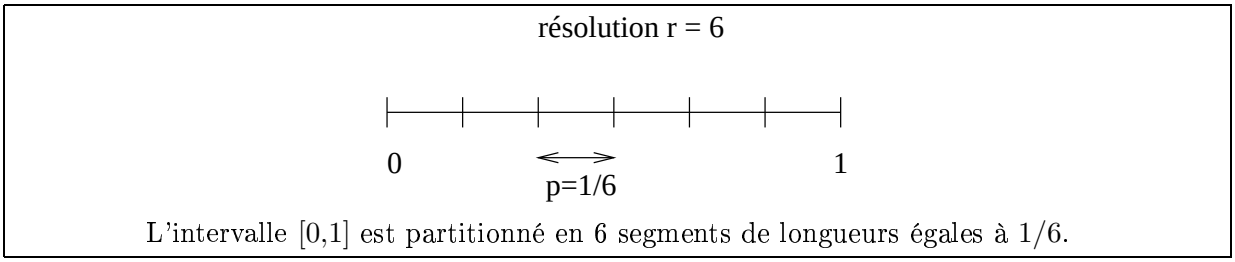


Fig. B.1 – L'intervalle [0,1] vu avec une résolution r=6.

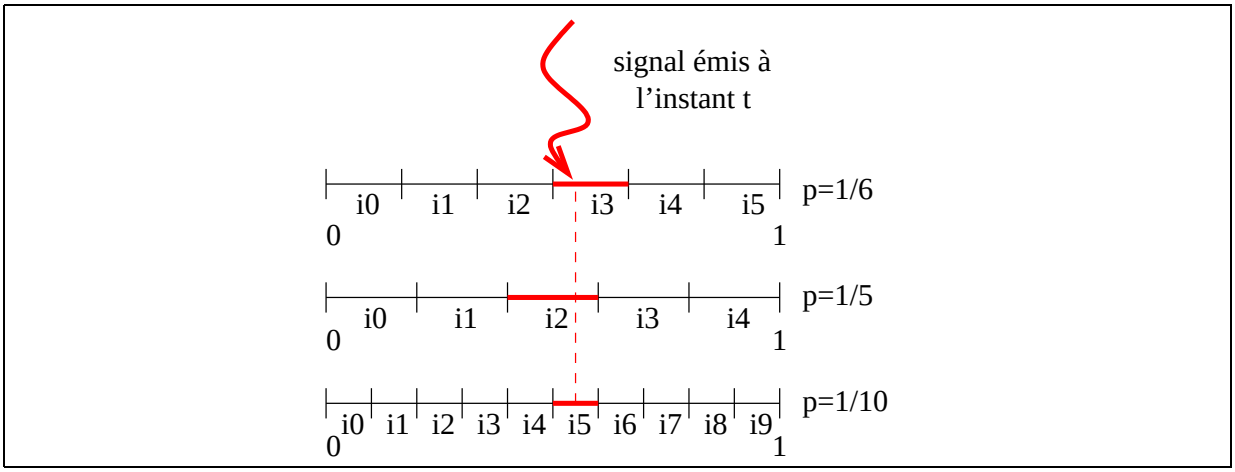


Fig. B.2 – Application du signal X, vue sous différentes résolutions

À partir d'une suite quelconque de segments, il est possible de créer une suite unique de segments orientés. Les éléments de cette suite, c'est-à-dire les segments orientés, forment les informations perceptives de base du système P_X. Celles-ci expriment donc une succession de mouvements, pour chacune des résolutions possibles.

B.1.2 Fiabilité des informations perceptives obtenues

Dans notre démarche, une information ne peut être utilisée dans le cadre d'un processus décisionnel que si elle est fiable. Ainsi, l'ensemble des données disponibles pour prendre une décision doivent être exactes, ce qui ne signifie pas qu'elles soient précises : notre objectif est de dégager les traits les plus fins possibles d'un signal, en ayant assurance de leur exactitude. Dans ce paragraphe, nous montrons qu'une tendance, prise isolément, n'est pas une information fiable. À partir de ce constat, nous indiquons comment la prise en compte d'informations supplémentaires

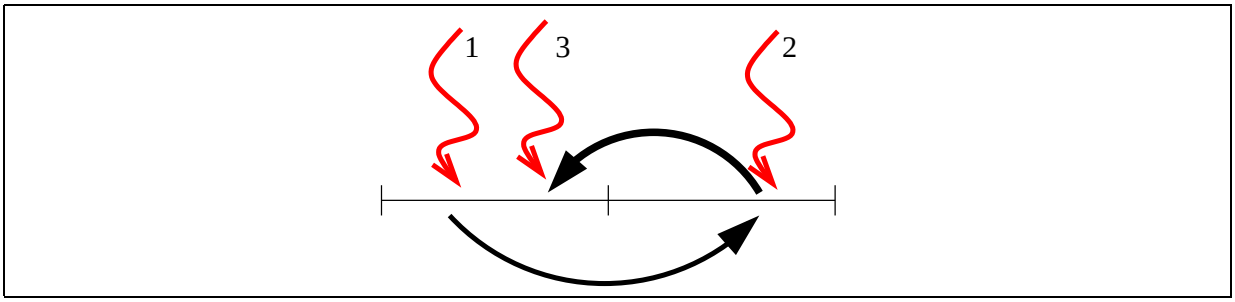
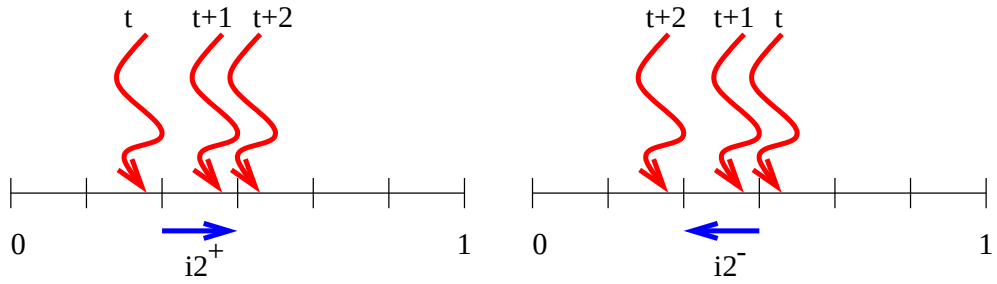
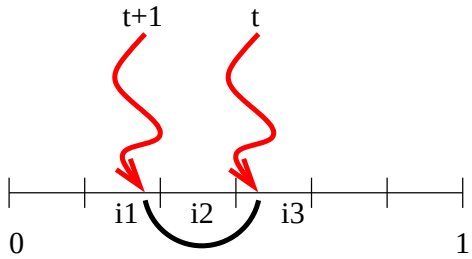


Fig. B.3 – La suite de signaux donne une tendance de retournement



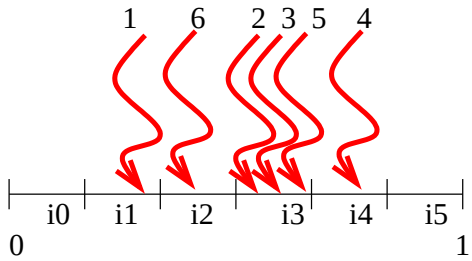
À gauche, le segment i_2 est orienté positivement et noté i_2^+, alors qu'à droite le segment i_2 est orienté négativement et noté i_2^-.

Fig. B.4 – Segments orientés.



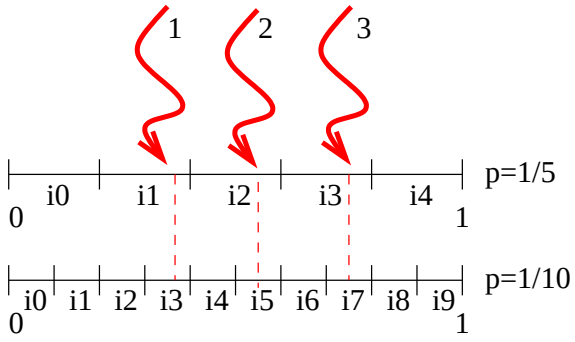
Les données $X(t)$ et $X(t+1)$ touchent respectivement les segments i_3 et i_1, qui ne sont pas voisins.

Fig. B.5 – Le signal X est incohérent pour une résolution r=6



La séquence des signaux 1,2,3,4,5 et 6 engendre la construction de la suite de segments touchés i_1,i_3,i_4,i_3,i_2. La première occurrence de i_3 apparaît lorsque la transition de i_1 vers i_3 est détectée (signal 2). Le signal 3, touchant également i_3 n'engendre pas la construction d'une deuxième occurrence de i_3 consécutive.

Fig. B.6 – Construction d'une suite de segments touchés par un signal



Pour une résolution de 5, l'évolution du signal 1,2,3 engendre la suite i_1,i_2,i_3 qui est cohérente. Par contre, pour une résolution 10, l'évolution du signal 1,2,3 produit la suite i_3,i_5,i_7 qui n'est pas cohérente.

FIG. B.7 – La notion de cohérence est dépendante de la résolution.

au cours du temps permet de confirmer ou d'infirmer l'information initiale. En outre, ce processus permet d'éliminer totalement la production d'informations à partir d'un signal aléatoire de probabilité uniforme sur [0,1].

Admettons qu'on détecte un segment orienté. Il est utile de se demander dans quelle proportion cette détection provient d'une réalité observable (le signal traverse réellement ce segment) ou d'un phénomène de « mirage » dû au bruit de mesure introduit par le capteur. Pour cela, nous allons considérer un signal aléatoire X régi par une loi uniforme sur [0,1]. Il paraît évident que ce signal ne doit dégager aucune cohérence et qu'on ne peut tirer aucune information fiable à partir de celui-ci. Pourtant, la probabilité Pr de découvrir une information de base (tendance positive ou négative) est non nulle est vaut :

$$Pr = Pr^+ + Pr^- = \frac{2}{(r-1)^2}, r \geq 3 \tag{B.1}$$

Avec Pr^+ (resp. Pr^-) représentant la probabilité de trouver par hasard une tendance positive (resp. négative) et r représentant la résolution avec laquelle le système P_X perçoit le signal aléatoire. Dans notre cas, r est supérieur ou égal à 3, car le nombre de segments doit être au moins de 3 pour pouvoir détecter une tendance. Cette relation est démontrée en annexe B.1.3, page 144 et vérifiée par l'expérience dans cette même annexe.

Ce résultat signifie, sans surprise, qu'on ne peut pas se fier à une tendance prise isolément pour une résolution donnée, car la survenue d'une erreur est très probable. Mais, quelles sont les possibilités permettant de réduire ce risque de manière à ce que l'apparition d'une erreur soit théoriquement possible mais pratiquement très improbable ? Car, dans le cas où nous pouvons trouver une méthode permettant de minimiser ce risque de manière à ce qu'une erreur ne surgisse jamais en réalité (même si elle est théoriquement possible), cela signifie que nous sommes capables de spécifier la nature d'informations qui, lorsqu'elles sont détectées, ne peuvent pas être mises en doute.
D'après l'équation B.1, la probabilité Pr_k^+ ou Pr_k^- pour que k tendances de même nature (positive ou négative) soient découvertes consécutivement dans le signal aléatoire X est :

$$Pr_k^+ = Pr_k^- = \frac{1}{(r-1)^{k+1}}, r \geq 2+k \tag{B.2}$$

De même, la probabilité Pr_k pour que k tendances (quelle que soit leur nature) soient découvertes consécutivement dans le signal aléatoire X est :

$$Pr_k = \left(\frac{2r^{k-1}}{(r-1)^{2k}} \right), r \geq 2 + k \tag{B.3}$$

En augmentant k, on peut diminuer les termes Pr_k^+, Pr_k^- et Pr_k autant qu'on le souhaite. En dessous d'un certain seuil ϵ, on pourra considérer que l'événement n'arrivera jamais en pratique (dans un temps « raisonnable »). Les équations B.2 et B.3 permettent de donner k^+,k^- et k en fonction de ϵ et r :

$$k^+ = k^- = \left[-1 - \frac{log(\epsilon)}{log(r-1)} \right] + 1, r \geq 3 \tag{B.4}$$

$$k = \left[\frac{log(\frac{1}{2}\epsilon r)}{log(\frac{r}{(r-1)^2})} \right] + 1, r \geq 3 \tag{B.5}$$

Avec [] désignant la fonction partie entière et *log* la fonction logarithme décimal. La table B.1 donne les valeurs de k, k^+ et k^- pour $\epsilon = 10^{-15}$ [1]. La démonstration de ces deux relations est présentée dans la section suivante. Il est clair que la valeur de ϵ détermine le degré de certitude dans le temps de l'information fournie. Pour une durée déterminée, cette valeur conditionne la probabilité pour qu'une erreur survienne, dans la mesure où on connaît le nombre d'informations obtenues dans cet intervalle de temps.

Les tendances sont des briques de base dans la perception du mouvement du signal. On peut les combiner avec des informations concernant la localisation du signal. Si on considère le signal aléatoire X de loi uniforme sur [0,1], la probabilité pour que $X(t)$ appartienne à un segment donné, pour une résolution r, est $1/r$. Ainsi, la probabilité Pr_p pour que p valeurs consécutives $X(t)$,$X(t+1)$,...$X(t+p-1)$ soient dans ce même segment est :

$$Pr_p = \left(\frac{1}{r} \right)^p, r \geq 3 \tag{B.6}$$

Comme précédemment, on peut définir une valeur de p telle que Pr_p soit inférieur ou égal à une probabilité seuil ϵ.

$$p = \left[-\frac{log(\epsilon)}{log(r)} \right] + 1, r \geq 3 \tag{B.7}$$

La table B.2 donne les valeurs de p pour $\epsilon = 10^{-15}$.

Un regard rapide sur les tables B.1 et B.2 nous montre que l'exigence de certitude, telle que nous l'avons définie, aboutit à des valeurs de k,k^+,k^- et p très élevées, voire même irréalisables pour k^+ et k^- avec les résolutions 5 et 10. L'expérience montre qu'avec ce degré d'exigence ($\epsilon = 10^{-15}$), aucune tendance ni aucun positionnement statique n'est détecté pour un signal aléatoire de loi uniforme sur [0,1].

Appliquons maintenant notre démarche de recherche de cohérence sur un signal Y de la forme $Y(t) = (1-\lambda)\, f(t) + \lambda X(t)$, avec $\lambda \in [0,1]$, $f(t) = \frac{1}{2}\,(1 + sin(t))$ et $X(t)$ signal aléatoire de densité de probabilité uniforme sur [0,1]. Nous constatons que lorsque l'amplitude de l'écart maximum entre deux points consécutifs $Y(t)$ et $Y(t+1)$ est inférieure strictement à la résolution,

1. Nous utiliserons cette valeur arbitraire de ϵ comme référence.

TAB. B.1 – Valeurs de k, k^+ et k^- pour $\epsilon = 10^{-15}$

Résolution r	5	10	25	50	100
k	29	16	11	9	7
k^+,k^-	21	15	10	8	7

Les calculs ont été effectués grâce aux équations B.4 et B.5.

la cohérence du signal, à ce niveau de résolution, n'est jamais rompue (tous les points du signal font partie d'une tendance) ; par contre, le nombre de fois où le signal est déclaré être cohérent décroît très rapidement si cette amplitude devient supérieure à la résolution (voir la figure B.8). Cette technique a l'avantage de ne produire des informations que pour des résolutions en concordance avec l'amplitude du bruit : les résolutions trop fines par rapport à celui-ci n'expriment quasiment aucune information. Cependant, elle implique une contrainte forte sur la nature du bruit de mesure : celui-ci doit être borné, d'amplitude inférieure à la taille d'un segment (dans notre cas, l'amplitude doit être impérativement inférieure à 0.2). Or, si le bruit de mesure n'est pas borné (bruit gaussien, par exemple), les performances obtenues précédemment s'effondrent. Pour illustrer ce fait, prenons un signal Y défini c omme suit :

$$\begin{cases} Y(t) = \frac{1}{2}\left(1 + sin(t)\right) \text{ ,si } C(t) < \lambda \\ Y(t) = X(t) \text{ ,si } C(t) \geq \lambda \end{cases}$$

Avec C et X deux variables aléatoires de loi uniforme sur [0,1]. La figure B.9 montre les résultats obtenus. Nous remarquons que plus la résolution est grossière (r peu élevé), plus les détériorations sont rapides. Cela est dû simplement au fait que plus un segment est grand, plus le nombre de points le touchant est important (suivant la fréquence de rafraîchissement du signal, exprimée par la constante τ définie au début du paragraphe B.1.1), donc plus la probabilité d'obtenir une donnée incohérente dans ce segment est élevée.
En conclusion, l'utilisation stricte de l'algorithme de recherche de cohérence permet d'éliminer en pratique les informations « mirage » qui pourraient résulter d'un bruit uniforme. Néanmoins, cet algorithme s'avère inadapté pour deux raisons :

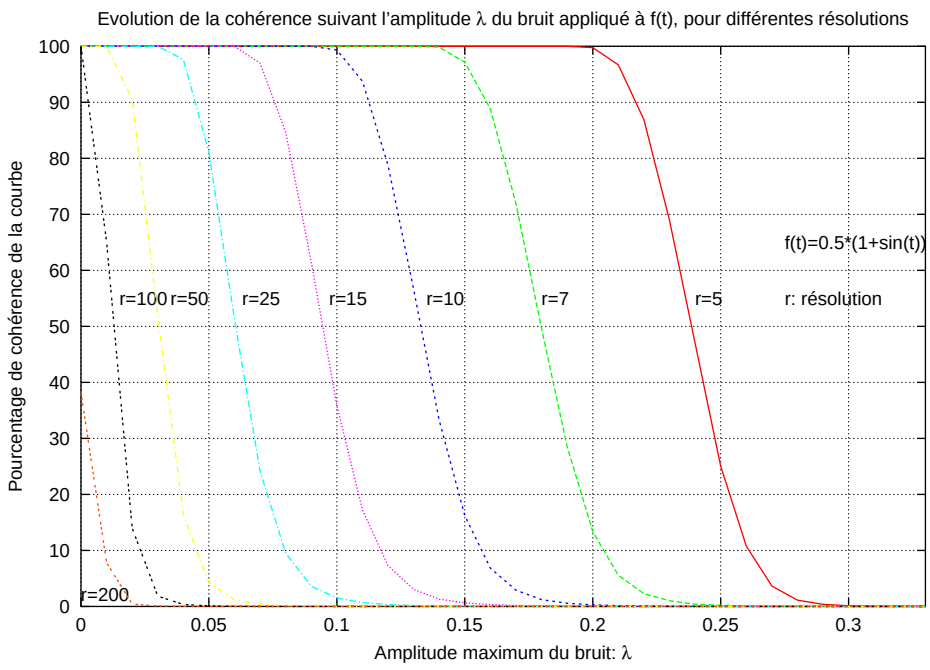
Problème 1 *: le nombre de confirmations consécutives est très grand. Par conséquent, lorsque le signal varie rapidement ou lorsque le taux d'échantillonnage est trop faible, le nombre de mesures incluses dans un segment devient trop faible pour apporter une information certaine, puisque le nombre de confirmations nécessaires n'est pas atteint. Le résultat est un « trou » dans la suite des informations que notre technique apporte. La figure B.10 illustre ce fait.*

Problème 2 *: lorsque l'amplitude maximum du bruit dépasse la taille d'un segment, la cohérence est rompue, même si la probabilité pour que l'écart entre deux mesures dépasse la taille de ce segment est très faible.*

Tab. B.2 – Valeurs de p pour $\epsilon = 10^{-15}$

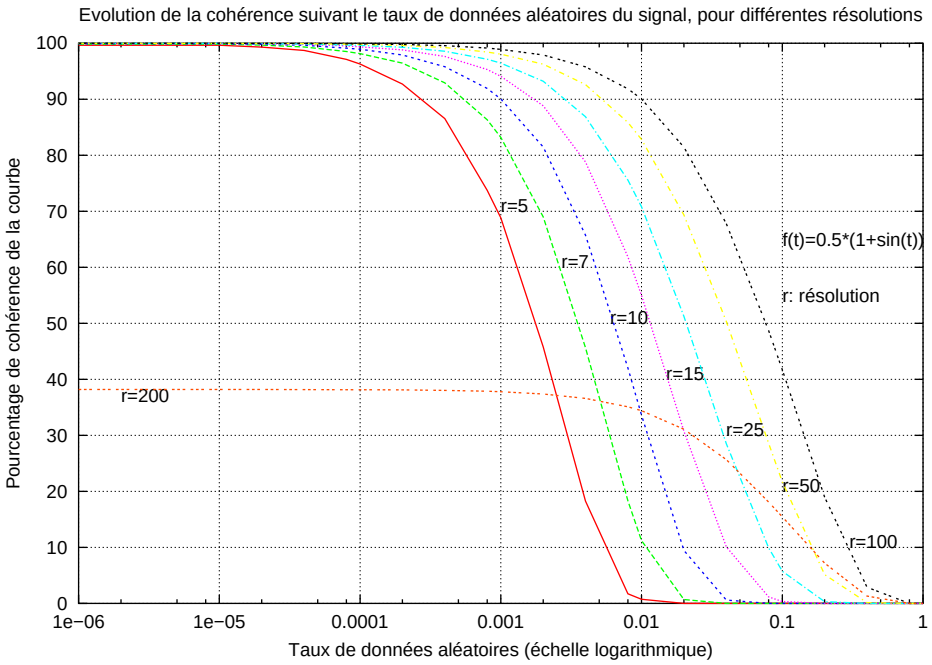
Résolution r	5	10	25	50	100
p	22	16	11	9	8

Les calculs ont été effectués grâce à l'équation B.7.



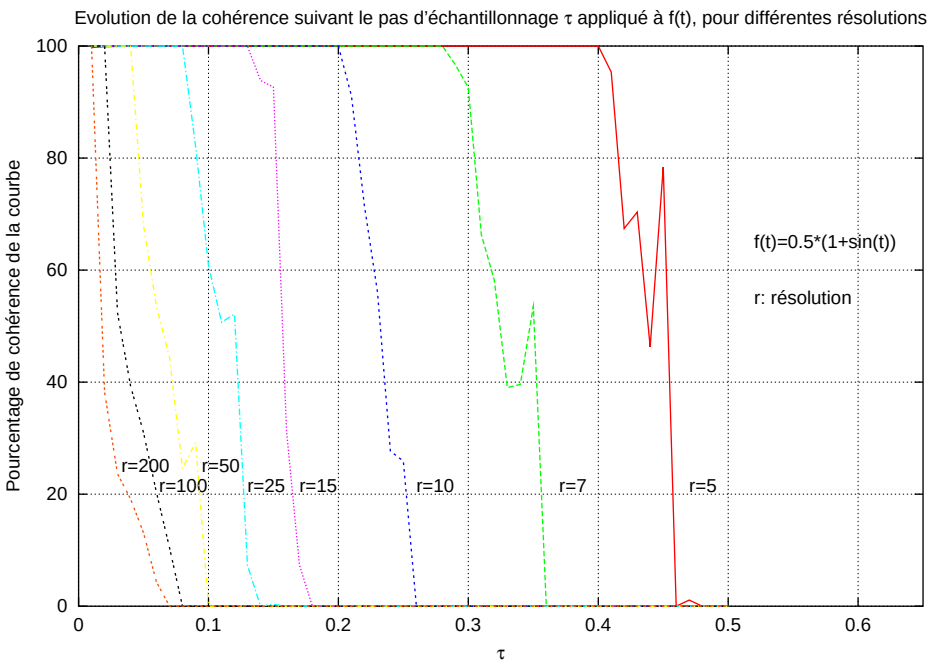
La fonction artificiellement bruitée par un bruit de loi uniforme d'amplitude λ est $f(t) = \frac{1}{2}(1 + sin(t))$.

Fig. B.8 – Cohérence détectée en fonction de l'amplitude du bruit de mesure



La fonction est $f(t) = \frac{1}{2}(1 + sin(t))$.

FIG. B.9 – Cohérence détectée en fonction du taux de données aléatoires



La fonction est $f(t) = \frac{1}{2}(1 + sin(t))$. Elle n'a pas été bruitée artificiellement.
On remarque que la cohérence commence à chuter lorsque $\tau > 2r$. On retrouve ainsi un exemple du résultat du théorème de Shannon sur l'échantillonnage.

FIG. B.10 – Cohérence détectée en fonction du taux d'échantillonnage.

B.1.3 Probabilité de découverte au hasard d'un segment orienté

Cette sous-section fait référence à l'équation B.1 donnée page 139 (pour le détail des notations, se reporter aux deux sous-sections précédentes) :

$$Pr = Pr^+ + Pr^- = \frac{2}{(r-1)^2}, r \geq 3$$

Soit X un signal dont la distribution suit une loi uniforme sur [0,1] et r la résolution avec laquelle le signal est vu. Le segment [0,1] est partitionné en r segments de même longueur $l = \frac{1}{r}$, nommés i_0,i_1,...,i_{r-1}.

Considérons un segment i_{k-1}, avec $k \in \{2,...,r-3\}$, touché à l'instant t par le signal $X(t)$. Nous souhaitons calculer la probabilité Pr^+ de détecter le segment orienté i_k^+. Le schéma de détection de ce segment orienté est donné par le graphe B.11.

On en déduit que la probabilité de détection effective du segment orienté i_k^+ au bout de j pas de temps s'exprime en fonction du passage dans les états k-1, k et k+1. Ainsi, pour 2 pas de temps (temps minimum de validation), la probabilité vaut l^2 (état k à l'instant t+1 et k+1 à l'instant t+2). Pour 3 pas de temps, la probabilité vaut $2l^3$ (état k-1 à l'instant t+1, état k à l'instant t+2 et état k+1 à l'instant t+3 OU état k aux instants t+1 et t+2, état k+1 à l'instant t+3). En général, pour j pas de temps, la probabilité vaut $(j-1)l^j$: le « j-1 » est obtenu en comptant le nombre de possibilité de trouver deux entiers (nombre de passages dans l'état k-1 et nombre de passages dans l'état k) tels que leur somme vaut j-1.

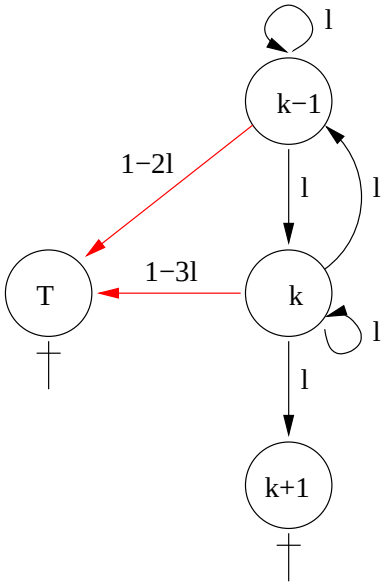
Par conséquent, $Pr^+ = \sum_{j=2}^{\infty}(j-1)l^j$. Cette expression existe, car l est strictement inférieur à 1. Elle peut se mettre sous la forme $l^2 \sum_{j=0}^{\infty}(j+1)l^j$. Le résultat de cette somme infinie est « classique » et vaut $\frac{1}{(1-l)^2}$. Par conséquent, $Pr^+ = \frac{l^2}{(1-l)^2}$. En divisant par l^2, on obtient l'expression désirée : $Pr^- = \frac{1}{(r-1)^2}$.

Le fait de découvrir une tendance (positive ou négative) étant la réunion de deux faits indépendants, dont les probabilités sont Pr^+ et Pr^-, on en déduit l'expression désirée de la probabilité de découvrir une tendance (positive ou négative).

Lorsque $k \in \{1,2\}$, la probabilité de découvrir une tendance négative à partir de i_{k-1} est nulle, puisqu'il faut au minimum trois sous-segments consécutifs. De même, si $k \in \{r-1,r-2\}$, la probabilité de découvrir une tendance positive est nulle.

Pour confirmer expérimentalement les expressions de Pr^+ et Pr^-, nous avons utilisé le processus de détection de cohérence en le soumettant à un signal $X(t)$ de densité de probabilité uniforme sur [0,1]. Nous avons compté le nombre de tendances négatives trouvées et calculé le ratio entre ce nombre et le nombre total de recherches de cohérence. Dans ce cadre, il faut noter que ce ratio n'est pas exactement égal à $Pr-$: en effet, une tendance négative ne peut pas être découverte si la recherche de cohérence débute sur l'un des deux premiers sous-segments i_0 ou i_1, car la détection nécessite trois sous-segments consécutifs. Par conséquent, la probabilité réelle de trouver une tendance négative à partir du signal (t) (tous sous-segments de départ confondus) est donnée par l'expression $(1 - \frac{2}{r}).Pr^-$.

Les résultats concernant les résolutions de 5 à 100 par pas de 5 sont synthétisés dans la figure B.12. Pour chaque résolution, l'algorithme est poursuivi jusqu'à atteindre 20000 détections de



Les arcs sont marqués par la probabilité de transition d'un état à l'autre. L'état de validation de la détection du segment orienté i_k^+ est « k+1 ». L'état d'invalidation de la détection est « T ».

FIG. B.11 – Graphe des évolutions possibles dans le processus de détection d'un segment orienté

segments orientés négativement. Un nombre d'itérations identique pour chacune des résolutions aurait donné des résultats moins significatifs pour des résolutions grandes, pour lesquelles un trop faible nombre de découvertes peut apparaître, rendant la comparaison théorie/expérience trop imprécise.

B.1.4 Découverte de n tendances consécutives pour un signal de densité de probabilité uniforme

Nous allons montrer, puis vérifier expérimentalement, la relation B.2, liant la probabilité Pr_k^+ de découvrir k tendances consécutives de même signe avec cette valeur k, lorsqu'un signal $X(t)$ de densité de probabilité uniforme sur [0,1] est appliqué :

$$Pr_k^- = \frac{1}{(r-1)^{k+1}}, r \geq 2+k$$

Nous montrons cette expression par récurrence sur k. Elle est vraie pour k=1 (relation B.1). Admettons qu'elle soit vraie pour $k > 1$. Soit i_k^+ la tendance positive que nous venons de découvrir « par hasard ». Or, lorsqu'on a découvert i_k^+, la probabilité d'en trouver une seconde consécutivement i_{k+1}^+ se résume à la probabilité de passer de l'état i_{k+1} à l'état i_{k+2} (en reprenant un graphe similaire à la figure B.12).

En reprenant le même raisonnement que pour la démonstration de l'expression de Pr^+ (voir l'annexe B.1.3), la probabilité de passage de i_{k+1} à i_{k+2} en un pas de temps est égale à $\frac{1}{r}$; pour deux pas de temps, la probabilité est de $\frac{1}{r^2}$; pour j pas de temps, la probabilité est de $\frac{1}{r^j}$. On en déduit que la probabilité de passage de i_{k+1} à i_{k+2} s'exprime par la somme infinie suivante : $\sum_{j=1}^{\inf} \frac{1}{r^j}$. Or, cette expression est égale à $\frac{1}{r-1}$. Par conséquent, la probabilité de détecter k+1 tendances positives consécutives est : $Pr_{k+1} = Pr_k . \frac{1}{(r-1)} = \frac{1}{(r-1)}$. Cette expression est valable

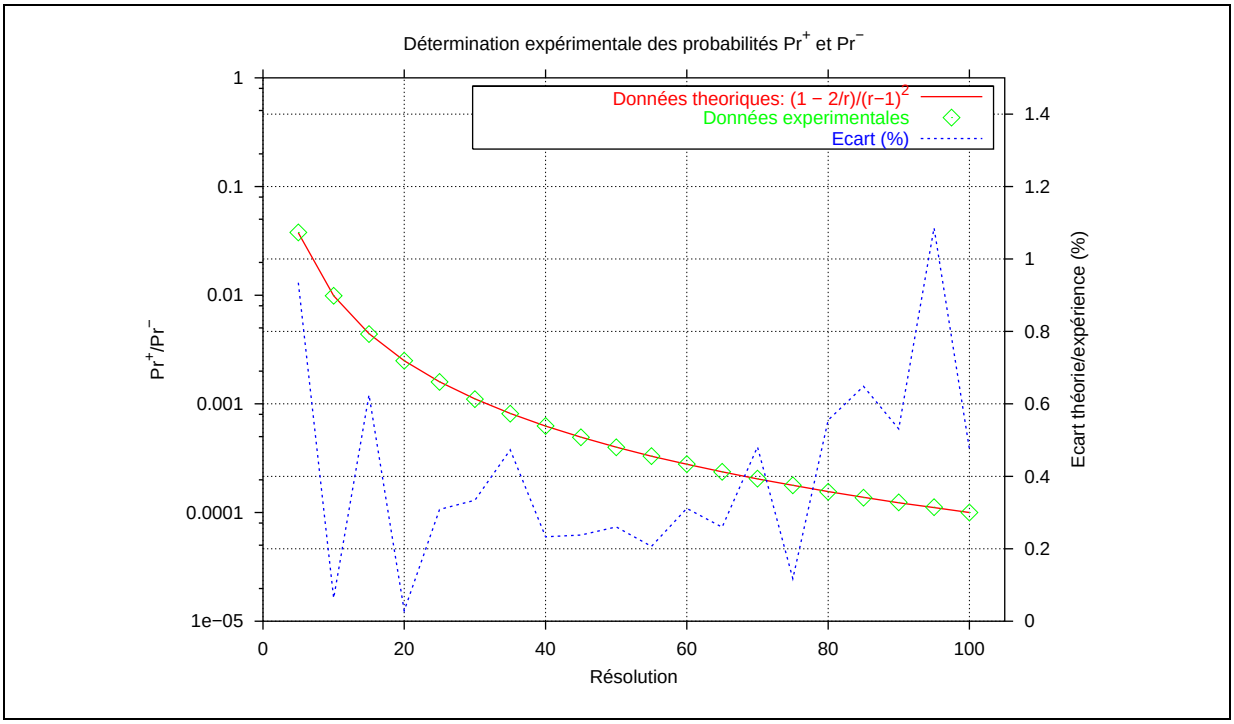


Fig. B.12 – Confrontation des résultats théoriques et expérimentaux concernant l'expression de Pr-

uniquement si l'ensemble des segments i_k comporte au moins un élément supplémentaire i_{k+2} par rapport au degré de récurrence inférieur. Donc, elle est valable lorsque le nombre de segments est supérieur ou égal à (k+1) + 1. Ce qui termine la récurrence.

En ce qui concerne la découverte de k tendances consécutives (quelle que soit la nature des tendances), la probabilité Pr s'exprime par la relation B.3 :

$$Pr_k = \left(\frac{2r^{k-1}}{(r-1)^{2k}} \right), r \geq 2 + k$$

Nous allons montrer cette relation par récurrence.

Constatons qu'elle est vraie pour k=1 (il s'agit la relation B.1). Admettons qu'elle soit vraie pour $k > 1$. Considérons la dernière tendance que nous venons de découvrir « par hasard ». Pour fixer les idées, nous admettrons que celle-ci est positive et se nomme i_k^+. Or, lorsqu'on a découvert i_k^+, la probabilité d'en trouver une seconde consécutivement découle de deux cas possibles : on découvre une tendance positive i_{k+1}^+ ou on découvre une tendance négative i_k^-. Or, la probabilité d'occurrence du premier cas est $\frac{1}{r-1}$ (d'après la démonstration précédente) et la probabilité d'occurrence de la deuxième possibilité est $\frac{1}{(r-1)^2}$ (probabilité de construire une tendance à partir du segment i_{k+1}). Par conséquent, la probabilité de découvrir k+1 tendances « au hasard » est donnée par la relation de récurrence : $Pr_{k+1} = Pr_k.(\frac{1}{r-1} + \frac{1}{(r-1)^2})$. Or, par hypothèse de récurrence, $Pr_k = \left(\frac{2r^{k-1}}{(r-1)^{2k}}\right)$ et $\frac{1}{r-1} + \frac{1}{(r-1)^2} = \frac{r}{(r-1)^2}$, ce qui donne la relation énoncée au rang k+1. D'autre part, si la nouvelle tendance découverte est positive, on a besoin d'un segment en plus par rapport aux hypothèses du rang k, c'est-à-dire $r \geq 2+k+1$. Ce qui termine la récurrence.

Après avoir effectué ces deux démonstrations, nous allons les confirmer expérimentalement. Pour cela, nous allons utiliser un dispositif expérimental similaire à celui du paragraphe B.1.3. Pour la confirmation expérimentale de l'expression B.2, nous allons utiliser une résolution r=10 et nous allons compter le nombre d'itérations de l'algorithme de suivi de cohérence pour que 2000 occurrences de une, deux, trois, quatre et cinq tendances consécutives apparaissent. Nous détectons uniquement des tendances positives en fixant la première valeur du signal $X(0)$ dans le segment i_0 : cela permet d'éviter les cas où la découverte de n tendances positives consécutives est rendue impossible car le rang du segment touché par $X(0)$ est supérieur à $r-k-2$. D'autre part, pour la confirmation expérimentale de B.3, nous fixons la première valeur du signal $X(0)$ dans le segment médian de [0,1], pour les mêmes raisons que précédemment, et nous utilisons une résolution r=15.

Pour des raisons de temps de calcul, nous limitons à 200 le nombre d'occurrences découvertes de six tendances consécutives.

Les résultats sont rassemblés dans la figure B.13 et confirment la validité des deux relations B.2 et B.3.

Si l'on choisit une valeur de n assez grande, on peut donc faire chuter Pr_k,Pr_k^+ et Pr_k^- aussi bas qu'on le souhaite. Pour donner un ordre d'idée, si cette grandeur est de 10^{-15} (valeur suffisamment faible pour représenter le caractère de rareté), associée à l'apparition de n tendances consécutives, la probabilité pour que cet événement se produise au bout de p itérations est égale à $1-(1-10^{-15})^p$. Pour que cette probabilité soit égale à 0.95, p doit dépasser 1.310^{15} itérations !

B.2 Relation entre le paramètre ϵ et le postulat de rareté de l'information perceptive

Les relations 4.1 et 4.2 (sous-section 4.2.2, page 118) permettent d'établir une table des triplets (h, i, l) valides, lorsque ϵ est fixé. Les valeurs de h, i et l dépendent donc de ϵ. Quelle valeur choisir pour ce paramètre? Pour répondre à cette question, il faut revenir sur la particularité de notre notion de *rareté* : un événement est rare s'il ne se produit pas en pratique, dans **la durée de l'expérience**. Nous supposons par là que, si la probabilité pour que l'événement se produise dans la plage de temps de l'expérience est trop faible, alors cet événement ne se produira pas en réalité. Il faut donc fixer ϵ par rapport à la durée de l'expérience ainsi qu'à un degré de confiance qui sera très proche de 0 en pratique.

Le paramètre ϵ représente la probabilité pour laquelle une des contraintes de CO ne serait pas respectée. Sa valeur découle de la tolérance à l'erreur qu'on admet pouvoir supporter sur une certaine durée D. La probabilité Pr_D pour qu'aucune « fausse » information ne soit générée est donnée par la relation :

$$Pr_D = (1-\epsilon)^R \tag{B.8}$$

Avec $R = \frac{D}{h.\tau}$ et τ représente la durée moyenne entre deux appels au processus de catégorisation. Si on pose $Pr_D = 1-\delta$, avec $\delta \in]0,1[$, on en déduit l'expression de ϵ :

$$\epsilon = 1-(1-\delta)^{\frac{1}{R}} \tag{B.9}$$

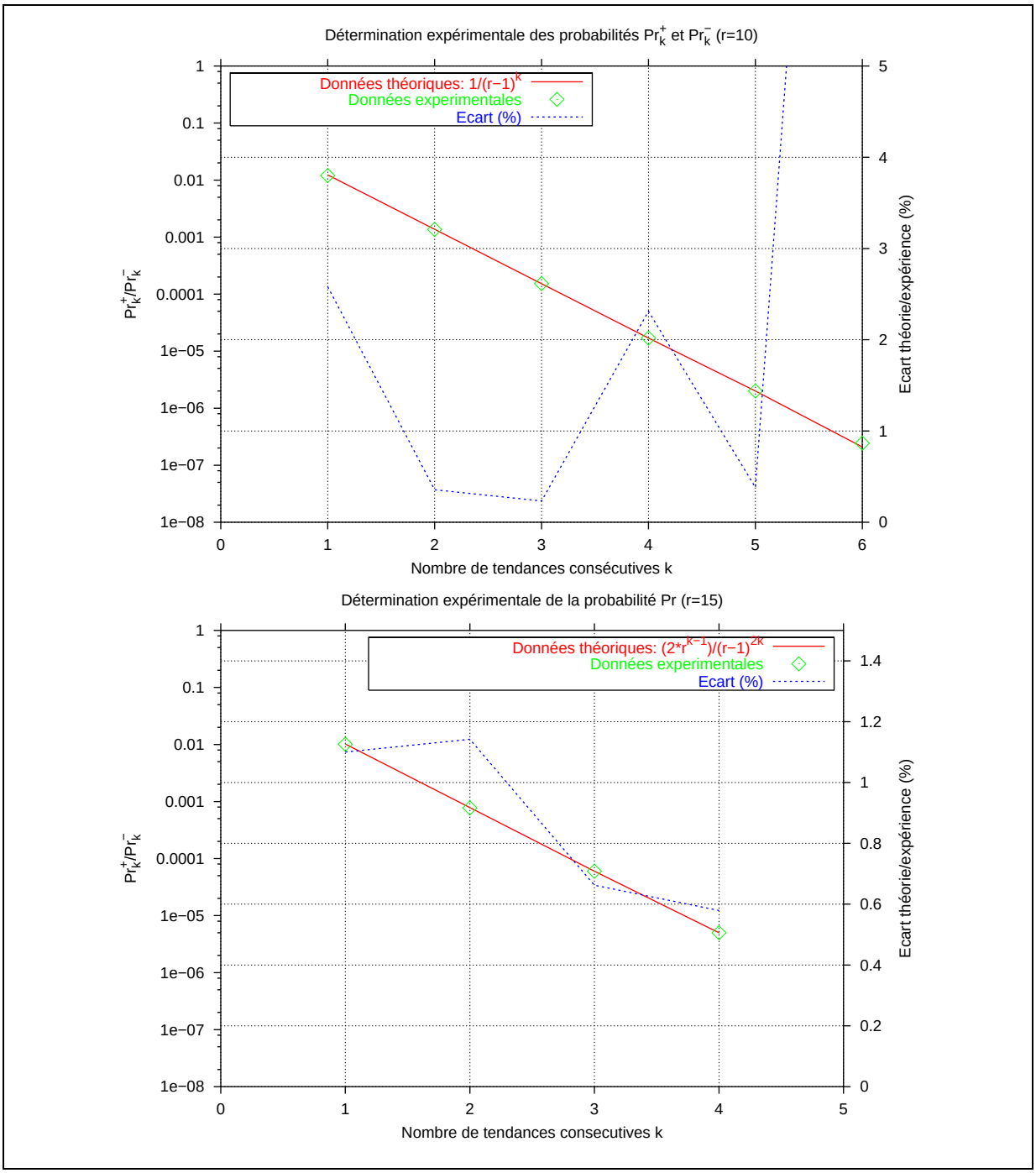


Fig. B.13 – Confrontation des résultats théoriques et expérimentaux concernant les expressions Prk et Prk+/-

Si on considère que δ est très inférieur à 1, l'équation précédente nous donne une forme approchée de ϵ :

$$\epsilon \sim \frac{\delta}{R} \tag{B.10}$$

Pour donner une idée, si on considère un problème pour lequel $h.\tau = 0.1sec$, D est égale à une année et $\delta = 10^{-6}$, on trouve $\epsilon \simeq 3.10^{-15}$. En revanche, si D vaut 1 sec et $\delta = 10^{-2}$, $\epsilon \simeq 10^{-3}$.

Les paramètres δ et D dépendent uniquement du contexte d'application dans lequel on utiliserait le processus de catégorisation.

B.3 Preuves concernant la fiabilité de la détection de l'information perceptive

B.3.1 Introduction - Notations

Cette section est consacrée à la preuve de la proposition 1, page 120. Nous allons procéder par étapes :

1. Montrer qu'il existe des triplets (h,i,l) tels que l'événement « Il n'y a pas de détection de l'information perceptive » est rare. C'est l'objet de la proposition 7, que nous prouverons dans la sous-section suivante.
2. Montrer qu'il existe des triplets (h,i,l) vérifiant CO. C'est l'objet de la proposition 8 (voir la sous-section B.3.3).
3. Donner **des conditions suffisantes** pour que l'intersection des deux ensembles soit non vide, ou pour que cette intersection soit vide, ce qui est l'objectif de cette section. Cela sera fait dans la sous-section B.3.4.

On considère le h-échantillon (b_1,b_2, ..., b_h) formé à partir des réalisations des variables aléatoires réelles B_1,B_2, ..., B_h. p_j est la probabilité pour que la réalisation b_j de la variable aléatoire B_j soit dans l'intervalle [-l/2,l/2]. Cette probabilité est identique à la probabilité que la valeur x_{t+j-1} du signal X à l'instant t+j-1 appartienne au focus. On considère la variable aléatoire discrète U_j, à valeurs dans {0,1}, qui vaut 1 si la réalisation b_j est dans l'intervalle [-l/2,l/2] et vaut 0 dans le cas contraire. U_j suit une loi de Bernoulli, de paramètre p_j. Enfin, on considère la variable aléatoire S, à valeurs discrètes dans {0,1,...,h}, qui représente le nombre de composantes de du h-échantillon (b_1,b_2, ..., b_h) comprises dans l'intervalle [-l/2,l/2] : $S = \sum_{j=1}^{h} U_j$.

Dans le cas où les p_j seraient égales, S suivrait une loi binomiale. Cependant, nous considérons que les p_j peuvent avoir des valeurs distinctes.

B.3.2 Fiabilité de la détection d'une information perceptive

Dans la démonstration qui suit, nous fixons la valeur de l. Il fait bien se rappeler que les p_j dépendent, par construction, de l.

Voici la proposition que nous allons montrer :

Proposition 7 *Si tous les p_j sont strictement positifs, alors il existe un ensemble de triplets (h,i,l) tel que l'événement « Il n'y a pas de détection de l'information perceptive » est rare. Cet ensemble possède une borne inférieure pour i qui dépend des p_j, telle que pour tout i supérieur à*

celle-ci, on peut trouver un h tel que (h,i,l) réponde à notre exigence de rareté. Lorsque i est fixé, l'ensemble des h possède une borne supérieure qui dépend également des p_j, telle que pour tout h inférieur à celle-ci, le triplet (h,i,l) répond à notre exigence de rareté.

▶

La notion de rareté sera interprétée à partir d'un réel $\epsilon \in]0,1[$ fixé *a priori*.

La probabilité d'occurrence de l'événement « Il n'y a pas de détection de l'information perceptive » est égale à P(S <h-i). Par définition, cet événement est déclaré rare si P(S <h-i) $<\epsilon$. C'est là notre point de départ.

P(S <h-i) s'écrit :

$$P(S \leq h-i) = \sum_{m=0}^{h-i} P(S=m)$$

Les probabilités $P(S \leq h-i)$ et $P(S=m)$ dépendent respectivement de h et i, puis de h et m. Nommons-les respectivement $u_{h,i}$ et $v_{h,m}$ et considérons les suites à deux indices $(u_{h,i})$ et $(v_{h,m})$.

Nous utilisons la propriété d'indépendance des B_j pour formuler l'expression générale de $v_{h,m}$:

$$v_{h,m} = \sum_{E \subset H(m)} \left(\prod_{j \in E} p_j \right) \left(\prod_{j \in \overline{E}} (1-p_j) \right)$$

Avec $H(m)$ ensemble des parties de $\{0,1,...,h-1\}$ comportant m éléments, et $\overline{E}$ ensembles des éléments de $\{0,1,...,h-1\}$ n'appartenant pas à E.

La démarche de notre démonstration est la suivante :

1. trouver une relation de récurrence liant les $v_{h,m}$
2. en déduire une relation de récurrence liant les $u_{h,i}$
3. à partir du point précédent, montrer que :
   (a) si on fixe *a priori* un $\epsilon \in]0,1]$, il existe un i_{min} tel que $u_{i_{min},i_{min}} \leq \epsilon$
   (b) la suite $(u_{h,i})$ est croissante suivant h, lorsque i est fixé, et elle tend vers 1 lorsque h tend vers l'infini.
   (c) à partir de ce dernier point, on en déduit l'existence de h_{max}.

Nous pouvons décomposer la somme $v_{h+1,m}$ en deux sommes, l'une comportant des ensembles E possédant le dernier élément h, et l'autre ne le possédant pas :

$$v_{h+1,m} = \sum_{E \subset H'(m), h \in E} \left(\prod_{j \in E} p_j \right) \left(\prod_{j \in \overline{E}} (1-p_j) \right) + \sum_{E \subset H'(m), h \notin E} \left(\prod_{j \in E} p_j \right) \left(\prod_{j \in \overline{E}} (1-p_j) \right)$$

Comme h fait partie de E dans la première expression de cette relation, le terme p_{h+1} fait partie du produit $\prod_{j \in E} p_j$ et on peut le mettre en facteur. De même, comme h ne fait pas partie de E dans la seconde expression, le terme $1-p_{h+1}$ peut être mis en facteur dans le produit

$\prod_{j\in\overline{E}}(1-p_j)$. Ce qui donne la relation suivante :

$$v_{h+1,m} = p_{h+1} \sum_{E\subset H'(m),h\in E} \left(\prod_{j\in E, j\neq h} p_j\right)\left(\prod_{j\in\overline{E}}(1-p_j)\right) + (1-p_{h+1}) \sum_{E\subset H'(m),h\notin E} \left(\prod_{j\in E} p_j\right)\left(\prod_{j\in\overline{E},j\neq h}(1-p_j)\right)$$

Or, l'expression $\sum_{E\subset H'(m),h\in E}\left(\prod_{j\in E,j\neq h} p_j\right)\left(\prod_{j\in\overline{E}}(1-p_j)\right)$ est exactement la même que $v_{h,m-1}$. De même, l'expression $\sum_{E\subset H'(m),h\notin E}\left(\prod_{j\in E} p_j\right)\left(\prod_{j\in\overline{E},j\neq h}(1-p_j)\right)$ vaut $v_{h,m}$. Ce qui donne le relation de récurrence suivante :

$$v_{h+1,m} = (1-p_{h+1})v_{h,m} + p_{h+1}v_{h,m-1} \tag{B.11}$$

Si on effectue la somme des éléments de la relation B.11 pour $m \in \{1,...,h-i\}$, on obtient la relation suivante :

$$\sum_{m=1}^{h-i} v_{h+1,m} = (1-p_{h+1})\sum_{m=1}^{h-i} v_{h,m} + p_{h+1}\sum_{m=1}^{h-i} v_{h,m-1}$$

Or, pour tout couple (h,i), avec $0 \leq i \leq h$, $\sum_{m=0}^{h-i} v_{h,m} = u_{h,i}$.
On en déduit les trois relations suivantes, par des changements d'indices appropriés :

$$\sum_{m=1}^{h-i} v_{h+1,m} = u_{h+1,i+1} - v_{h+1,0}$$

$$\sum_{m=1}^{h-i} v_{h,m} = u_{h,i} - v_{h,0}$$

$$\sum_{m=1}^{h-i} v_{h,m-1} = u_{h,i+1}$$

Or, $v_{h+1,0} = \prod_{j\in\{0,..,h\}}(1-p_{j+1}) = (1-p_{h+1})v_{h,0}$. Ce qui donne la relation :

$$u_{h+1,i+1} = (1-p_{h+1})u_{h,i} + p_{h+1}u_{h,i+1} \tag{B.12}$$

Comme $u_{h,i} = u_{h,i+1} + v_{h,i+1}$ et que $v_{h,i+1}$ est un terme positif ou nul, on en déduit que $u_{h,i+1} \leq u_{h,i}$. Or, p_{h+1} est une probabilité, donc est compris dans l'intervalle [0,1]. Par conséquent, la relation B.12 est une relation barycentrique et implique l'inégalité suivante :

$$\forall(h,i), h \geq i \geq 0, u_{h,i+1} \leq u_{h+1,i+1} \leq u_{h,i}$$

On en déduit que lorsque i est fixé, strictement positif, la suite $(u_{h,i})$ est croissante suivant h. Par conséquent, $\forall h \geq i, u_{h,i} \geq u_{i,i}$. Or, $u_{i,i} = \prod_{j=1}^{i}(1-p_j)$. Comme on suppose, par hypothèse, que l'ensemble des p_j peut être minoré par un p_{min} strictement positif (ce qui signifie que la probabilité d'être à l'intérieur d'un segment de largeur l n'est jamais nulle), $u_{i,i}$ est strictement décroissante et tend vers 0 lorsque i tend vers l'infini. Ce qui signifie que si on fixe *a priori* un

$\epsilon \in]0,1]$, il existe un i_{min} tel que $u_{i_{min},i_{min}} \leq \epsilon$.
D'autre part, si on choisit un $i \geq i_{min}$, on a également $u_{i,i} \leq \epsilon$, d'après la remarque sur la décroissance de $(u_{i,i})$.
Nous allons à présent montrer que la suite $(u_{h,i})$ tend vers 1 suivant h, lorsque i est fixé.
Nous avons déjà remarqué que la suite $(u_{h,i})$ est croissante suivant h, lorsque i est fixé. Or, les $u_{h,i}$ sont des probabilités, donc $(u_{h,i})$ est majorée par 1. Par conséquent, $(u_{h,i})$ converge suivant h, lorsque i est fixé. Soit $l_i = \lim_{h\to\infty} u_{h,i}$ et $l_{i+1} = \lim_{h\to\infty} u_{h,i+1}$. La convergence ainsi que la croissance de $u_{h,i}$ suivant h, lorsque i est fixé, permettent d'affirmer :

$$\forall \alpha > 0, \exists h_0 \in \mathbb{N}, \forall h \geq h_0, l_i - u_{h,i} < \alpha \; et \; l_{i+1} - u_{h,i+1} < \alpha$$

Comme $(u_{h,i+1})$ est décroissante suivant i, à h fixé, $u_{h,i+1} \leq u_{h,i}$, $l_i - u_{h,i} < \alpha \Rightarrow l_i - u_{h,i+1} < \alpha$. Or, d'après la relation B.12, $u_{h,i+1} \in [u_{h-1,i+1}, u_{h-1,i}]$. Par conséquent, $l_i - u_{h,i+1} \in [l_i - u_{h-1,i}, l_i - u_{h-1,i+1}]$. Comme $l_i - u_{h-1,i} \geq 0$, on en déduit l'encadrement :

$$0 \leq l_i - u_{h,i+1} < \alpha$$

En d'autres termes, cela étant vrai pour tout α, $l_i = \lim_{h\to\infty} u_{h,i+1} = l_{i+1}$.

Nous venons donc de montrer que pour tout i fixé, $\lim_{h\to\infty} u_{h,i} = l_i = l_{i-1} = ... = l_0$.
Or, $u_{h,0} = 1$ pour tout h. Ce qui montre $\lim_{h\to\infty} u_{h,i} = 1$ pour tout i.

Par conséquent, pour i fixé tel que $i \geq i_{min}$, il existe un h_{max} tel que :

$$\forall h \geq h_{max}, u_{h,i} \leq \epsilon \; et \; u_{h_{max}+1,i} > \epsilon$$

Nous venons donc de délimiter l'ensemble des triplets (h,i,l) pour lesquels l'événement « Il n'y a pas de détection de l'information perceptive » est rare.
◀

B.3.3 Preuve de la proposition 8

Voici la proposition que nous souhaitons montrer.

Proposition 8 *L'ensemble des triplets (h,i,l) permettant au système de respecter CO est non vide. Cet ensemble peut être caractérisé de la manière suivante : pour tout l fixé, l'ensemble des (h,i) permettant le respect de CO est non vide. De plus les valeurs de h de cet ensemble possèdent une borne inférieure dépendant de l, telle que pour tout h supérieur à celle-ci, on peut trouver un i pour que (h,i,l) respecte CO.*

▶
Nous reprendrons les notations utilisées dans la sous-section précédente.

La contrainte CO impose que la détection d'une information perceptive en sortie du système soit un événement rare. Nous considérons un signal X dont chaque B_j suit une loi uniforme sur [0,1] et le X_d vaut 0. Nous savons que le volume de l'ensemble des h-échantillons $b_1, b_2, ..., b_h$ générés à partir de X est égal à 1, qui correspond au volume de l'ensemble des h-échantillons générés à partir de l'ensemble de tous les signaux possibles. Comme la contrainte CO peut se

traduire en termes de calcul de volume de l'ensemble des solutions en sortie du système, on en déduit que notre démonstration ne changera pas si on considère un unique signal X, associé aux B_j que nous venons de définir. Or, la probabilité d'occurrence de l'événement « Une information perceptive est détectée » est égale à $\mathrm{P}(S \geq h-i)$. Dans le cas particulier de ce X, les p_j obtenus à partir des B_j sont égales et valent l. Le calcul de $\mathrm{P}(S \geq h-i)$ est donc facilité par rapport à la sous-section précédente, car la loi de S est binomiale, de paramètres h et l. Il vient :

$$Pr(S \geq h-i) = \sum_{m=h-i+1}^{h} C_h^m l^m (1-l)^{h-m}$$

Considérons la suite $(v'_{h,m})$ définie par la relation suivante :

$$\forall m \in \{1,..,h\},\ v'_{h,m} = C_h^m l^m (1-l)^{h-m}$$

Nous définissons la suite $(u'_{h,i})$ en fonction des $v'_{h,m}$:

$$\forall i \in \{1,..,h\},\ u'_{h,i} = \sum_{m=h-i+1}^{h} v'_{h,m}$$

Donc, $u'_{h,i} = Pr(T_S \geq h-i)$.
Or, sachant que $\sum_{m=0}^{h} v'_{h,m} = 1$, on peut transformer l'expression de $u'_{h,i}$:

$$u'_{h,i} = 1 - \sum_{m=0}^{h-i} v'_{h,m}$$

Or, la suite $(v'_{h,m})$ est un cas particulier de l'étude faite au paragraphe B.3.2, avec des p_j tous égaux et valant l . Lorsque i est fixé, on en déduit que :

$$lim_{h\to\infty} \sum_{m=0}^{h-i} v'_{h,m} = 1$$

Par conséquent, lorsque i est fixé :

$$lim_{h\to\infty} u'_{h,i} = 0$$

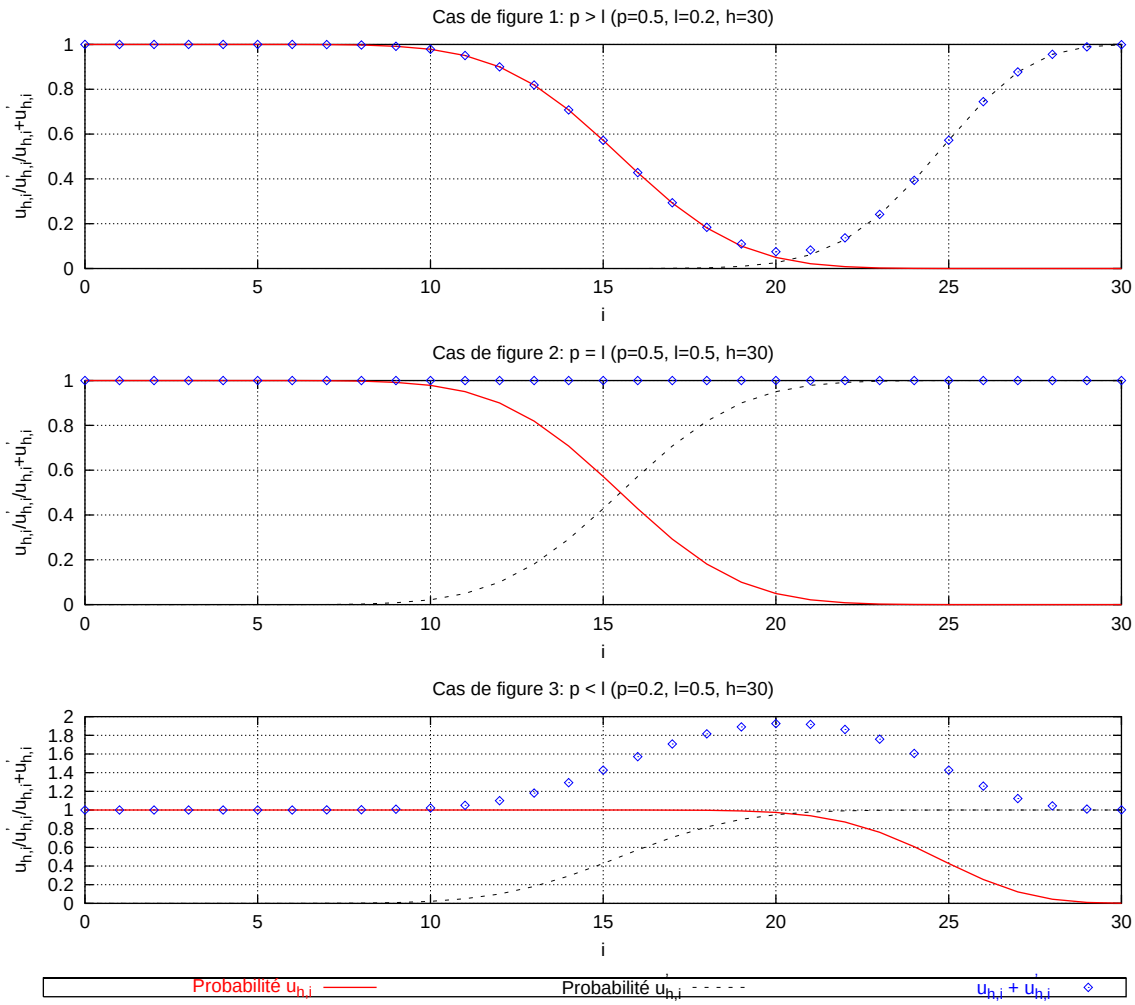
Par conséquent, lorsque $\epsilon > 0$ et i sont fixés, il existe un h_{min} tel que pour tout $h \geq h_{min}$, $Pr(T_S \geq h-i) < \epsilon$.
◀

B.3.4 Preuve du théorème 1 (paragraphe 4.2.3, page 120)

Dans ce paragraphe, nous utiliserons des notations identiques à celles employées dans les preuves données au cours des paragraphes B.3.2 et B.3.3, en particulier pour les suites $(u_{h,i})$, $(u'_{h,i})$ et $v_{h,m}$.

Dans un premier temps, nous allons considérer que les p_j sont tous égaux et valent p. Afin de mieux comprendre la démonstration qui suit, la figure B.14 montre l'évolution des suites $u_{h,i}$ et $u'_{h,i}$ suivant i, lorsque h est fixé, pour les trois cas de figure suivants :

1. l <p : cas où il existe des couples solution *h,i* pour tout $\epsilon > 0$.



La somme $u_{h,i} + u'_{h,i}$ représente la propension des deux suites à pouvoir ou non avoir des valeurs petites simultanément.

Fig. B.14 – Une idée de l'évolution des probabilités u et u' suivant i.

2. l = p : cas lié, entre autres, au fait de recevoir un signal $X(t)$ dont chacune des valeurs suit une loi uniforme sur [0,1], pour lequel il n'existe pas de couples solution.
3. l >p : cas lié, entre autres, au fait que le centre du focus est trop éloigné du point de densité maximale de $X(t)$.

Le plan de notre démonstration est le suivant :

1. Étude de l'évolution de $v_{h,m}$ suivant m, à h fixé.
2. Étude du cas l <p : on déduit du point précédent deux fonctions majorantes (une pour $u_{h,i}$ et une pour $u'_{h,i}$), qui tendent toutes les deux vers 0 lorsque h tend vers l'infini.
3. Étude du cas l = p : on montre que $u_{h,i} + u'_{h,i} = 1$ pour tout h et tout i.
4. Étude du cas l >p : on montre que $u_{h,i} + u'_{h,i} \geq 1$ pour tout h et tout $i \leq h$.

▶

Lorsque les p_j sont supposés constants, valant p, la probabilité $v_{h,m}$ s'exprime par une loi

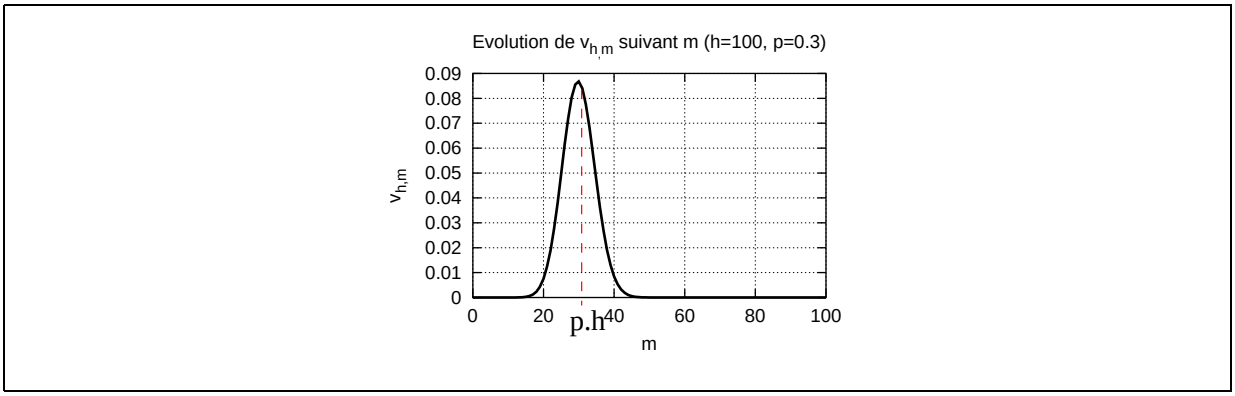


FIG. B.15 – Évolution de la suite v, suivant m, à h fixé.

binomiale de paramètres (h,m) :

$$v_{h,m} = C_h^m p^m (1-p)^{h-m}$$

Au regard de quelques exemples d'évolution de $v_{h,m}$ suivant m (voir la figure B.15), à h fixé, nous nous apercevons que l'élément maximum de cette suite est celui dont la valeur est la plus proche du produit $p.h$. Nous allons montrer que cette observation est justifiée.
Pour cela, nous utilisons une fonction auxiliaire prolongeant $v_{h,m}$ pour des valeurs non entières de h et m.

$$f(h,m) = c(h,m)p^m(1-p)^{h-m}$$

avec

$$c(h,m) = \frac{\Gamma(h)}{\Gamma(m)\Gamma(h-m)}$$

La fonction Γ d'Euler prolonge la fonction factorielle pour des valeurs non entières.
La dérivée partielle de c(h,m) suivant m est donnée par l'expression suivante :

$$\forall h > 0, m \in]0,h[,\ \frac{\partial c}{\partial m}(h,m) = c(h,m)\left(\frac{\Gamma'(h-m)}{\Gamma(h-m)} - \frac{\Gamma'(m)}{\Gamma(m)}\right)$$

Considérons la fonction d suivante :

$$\forall m \in [0,h[,\ d(m) = \ln(\Gamma(h-m)) + \ln(\Gamma(m))$$

La fonction d est continue et dérivable sur [1,h]. On remarque que la dérivée $d'(m) = -\frac{\Gamma'(h-m)}{\Gamma(h-m)} + \frac{\Gamma'(m)}{\Gamma(m)}$ Par conséquent, l'expression de la dérivée partielle de $c(h,m)$ suivant m s'écrit :

$$\forall h > 0, m \in [0,h[,\ \frac{\partial c}{\partial m}(h,m) = -c(h,m)d'(m)$$

On en déduit l'expression de la dérivée partielle de f suivant m :

$$\forall h > 0, m \in [0,h[,\ \frac{\partial f}{\partial m}(h,m) = f(h,m)\left[-d'(m) + ln\left(\frac{p}{1-p}\right)\right] \tag{B.13}$$

Comme f ne s'annule pas, l'annulation de $\frac{\partial f}{\partial m}(h,m)$ impose :

$$-d'(m) + ln\left(\frac{p}{1-p}\right) = 0$$

Nous allons à présent montrer que ce terme s'annule pour une valeur de m « proche » de ph. Par construction de Γ, on a :

$$\Gamma(ph+1) = (ph+1)\Gamma(ph)$$
$$\Gamma(h-ph) = (1-p)h\,\Gamma(h-(ph+1))$$

Par conséquent :

$$d(ph+1) - d(ph) = \ln(ph+1) - \ln((1-p)h)$$

En mettant p en facteur dans le premier terme de la différence :

$$d(ph+1) - d(ph) = \ln(p) - \ln(1-p) + \ln\left(\frac{h+1/p}{h}\right)$$

On a donc :

$$\forall h > 0, d(ph+1) - d(ph) > \ln\left(\frac{p}{1-p}\right)$$
$$lim_{h\to\infty}(d(ph+1) - d(ph)) = \ln\left(\frac{p}{1-p}\right)$$

On calcule de même :

$$d(ph) - d(ph-1) = \ln(p) - \ln(1-p) - \ln\left(\frac{h+1/(1-p)}{h}\right)$$

Par conséquent :

$$\forall h > 1, d(ph) - d(ph-1) < \ln\left(\frac{p}{1-p}\right)\, lim_{h\to\infty}(d(ph) - d(ph-1)) = \ln\left(\frac{p}{1-p}\right)$$

Or, d'après le théorème de Rolle, comme d est continue et dérivable sur [0,h[:

$$\exists m_1 \in [ph,ph+1], d'(m_1) = d(ph+1) - d(ph) \geq \ln(p) - \ln(1-p)$$
$$\exists m_2 \in [ph-1,ph], d'(m_2) = d(ph) - d(ph-1) \leq \ln(p) - \ln(1-p)$$

La dérivée d' étant continue, cela signifie que :

$$\forall h > 1, \exists m_0 \in [ph-1,ph+1], d'(m_0) = \ln(p) - \ln(1-p)$$

Ce qui montre que $\frac{\partial f}{\partial m}(h,m)$ s'annule sur $[ph-1,ph+1]$ pour tout h>1.
D'autre part, $\ln(\Gamma(m))$ étant une fonction convexe deux fois dérivable, sa dérivée seconde est une fonction strictement positive sur [1,h[. De même, la dérivée seconde de $\ln(\Gamma(h-m))$ est strictement positive sur [1,h[(par composition des fonctions $\Gamma(m)$ et $h-m$). Par conséquent, d' est croissante sur [1,h[. En particulier, on en déduit que d' ne prend la valeur $\ln(p) - \ln(1-p)$ qu'une seule fois, donc, en revenant à l'équation B.13 :

1. $\frac{\partial f}{\partial m}(h,m)$ ne s'annule qu'une seule fois, dans l'intervalle $[ph-1,ph+1]$.
2. d' étant croissante sur [1,h[et f positive, on en déduit que le point d'annulation de la dérivée partielle de f est un maximum (dérivée partielle positive sur [1,ph-1] et négative sur [ph+1,h[.

Ce qui conclut le premier point de la preuve.

Nous allons à présent étudier le comportement conjoint des suites $(u_{h,i})$ et $(u'_{h,i})$ suivant les trois cas évoqués au début de ce paragraphe.

Dans un premier temps, nous considérons l'hypothèse l <p. Le travail effectué dans le premier point nous permet d'écrire l'inégalité suivante, pour tout $k_p < [ph]$:

$$\forall k \leq k_p,\ C_h^k p^k (1-p)^{h-k} \leq C_h^{k_p} p^{k_p} (1-p)^{h-k_p} \tag{B.14}$$

De même, pour tout $k_l > [lh] + 1$:

$$\forall k \geq k_l,\ C_h^k l^k (1-l)^{h-k} \leq C_h^{k_l} p^{k_l} (1-l)^{h-k_l} \tag{B.15}$$

Par conséquent, en sommant suivant différentes valeurs de k, on obtient :

$$\sum_{k=0}^{k_p} C_h^k p^k (1-p)^{h-k} \leq (k_p+1) C_h^{k_p} p^{k_p} (1-p)^{h-k_p}$$
$$\sum_{k=k_l}^{h} C_h^k l^k (1-l)^{h-k} \leq (h-k_l+1) C_h^{k_l} l^{k_l} (1-l)^{h-k_l}$$

Or, pour tout p et tout l vérifiant p >l, on a la relation suivante :

$$\exists h_0, \forall h > h_0, \exists k_*,\ [lh] + 1 < k_* < [ph]$$

Par conséquent, k_* vérifie simultanément les relations B.14 et B.15. Or, si on pose $k_* = h - i_*$, avec $i_* \in [(1-p)h, (1-l)h]$, les deux sommes précédentes ne sont autres que u_{h,i_*} et u'_{h,i_*}. D'où les relations suivantes, vérifiées simultanément par i_* :

$$u_{h,i_*} \leq (h - i_* + 1) C_h^{i_*} p^{h-i_*} (1-p)^{i_*}$$
$$u'_{h,i_*} \leq (i_* + 1) C_h^{i_*} l^{h-i_*} (1-l)^{i_*}$$

Pour montrer que les deux majorants des relations B.14 et B.15 tendent vers 0, nous allons rechercher un équivalent de ceux-ci lorsque h tend vers l'infini. D'après la formule de Stirling, un équivalent de n ! est :

$$n! \underset{h\to\infty}{\sim} \sqrt{2\pi}\, n \left(\frac{n}{e}\right)^n$$

Nous remarquons que lorsque h devient grand, i_* devient grand, car $i_* \geq (1-p)h$. Par conséquent, l'équivalent précédent s'applique aux trois factorielles de $C_h^{i_*}$. On peut donc donner un équivalent du majorant de u_{h,i_*} et de celui de u'_{h,i_*}.

$$(h - i_* + 1) C_h^{i_*} p^{h-i_*} (1-p)^{i_*} \underset{h\to\infty}{\sim} \frac{h - i_* + 1}{\sqrt{2\pi}} \frac{h^{h+1}}{(h-i_*)^{h-i_*+1}\, i_*^{i_*+1}} p^{h-i_*} (1-p)^{i_*}$$
$$(i_* + 1) C_h^{i_*} l^{h-i_*} (1-l)^{i_*} \underset{h\to\infty}{\sim} \frac{i_* + 1}{\sqrt{2\pi}} \frac{h^{h+1}}{(h-i_*)^{h-i_*+1}\, i_*^{i_*+1}} l^{h-i_*} (1-l)^{i_*}$$

En posant $i_* = \alpha h$, avec $\alpha \in [1-p,1-l]$, les équivalents précédents s'écrivent :

$$\frac{h(1-\alpha)+1}{\sqrt{2\pi}\, h\alpha(1-\alpha)}\, g_p(h,\alpha)$$
$$\frac{h\alpha+1}{\sqrt{2\pi}\, h\alpha(1-\alpha)}\, g_l(h,\alpha)$$

avec :

$$g_p(h,\alpha) = \left(\frac{p}{1-\alpha}\right)^{(1-\alpha)h} \left(\frac{1-p}{\alpha}\right)^{\alpha h}$$
$$g_l(h,\alpha) = \left(\frac{l}{1-\alpha}\right)^{(1-\alpha)h} \left(\frac{1-l}{\alpha}\right)^{\alpha h}$$

Montrons à présent que $g_p(h,\alpha)$ et $g_l(h,\alpha)$ deviennent conjointement aussi petits qu'on le souhaite. Pour cela, considérons $\ln(g_p(h,\alpha))$ et $\ln(g_l(h,\alpha))$:

$$\ln(g_p(h,\alpha)) = hq_p(\alpha) \;\; avec\, q_p(\alpha) = (1-\alpha)\ln\left(\frac{p}{1-\alpha}\right) + \alpha\ln\left(\frac{1-p}{\alpha}\right)$$
$$\ln(g_l(h,\alpha)) = hq_l(\alpha) \;\; avec\, q_l(\alpha) = (1-\alpha)\ln\left(\frac{l}{1-\alpha}\right) + \alpha\ln\left(\frac{1-l}{\alpha}\right)$$

Les variations de $q_p(\alpha)$ et de $q_l(\alpha)$ sont données à partir de leur dérivée :

$$q_p^{'}(\alpha) = \ln\left(\frac{(1-p)(1-\alpha}{p\alpha)}\right)$$
$$q_l^{'}(\alpha) = \ln\left(\frac{(1-l)(1-\alpha}{l\alpha)}\right)$$

Or, si $\alpha \in [1-p,1-l]$, des encadrements montrent que $q_p^{'}(x) \leq 0$ et $q_l^{'}(x) \geq 0$ pour $x \in [1-p,1-l]$.
Par conséquent, $q_p(x) \leq q_p(1-p)$ et $q_l(x) \leq q_l(1-l)$. Or, $q_p(1-p) = q_l(1-l) = 0$.
Par conséquent, pour $\alpha \in]1-p,1-l[$, on a les deux inégalités $q_p(\alpha) < 0$ et $q_l(\alpha) < 0$, simultanément, indépendamment de h. Ce qui montre que :

$$\lim_{h\to\infty} \ln(g_p(h,\alpha)) = -\infty$$
$$\lim_{h\to\infty} \ln(g_l(h,\alpha)) = -\infty$$

Par conséquent :

$$\lim_{h\to\infty} g_p(h,\alpha) = 0$$
$$\lim_{h\to\infty} g_l(h,\alpha) = 0$$

Comme :

$$\lim_{h\to\infty} \frac{h(1-\alpha)+1}{\sqrt{2\pi}\, h\alpha(1-\alpha)} = \frac{1}{\sqrt{2\pi}\,\alpha} > 0$$

et

$$\lim_{h\to\infty} \frac{h\alpha+1}{\sqrt{2\pi}\, h\alpha(1-\alpha)} = \frac{1}{\sqrt{2\pi}\,(1-\alpha)} > 0$$

On en déduit que les fonctions majorantes de $(u_{h,i})$ et de $(u'_{h,i})$ peuvent être rendues simultanément aussi petites qu'on le souhaite, en fixant un i dans l'intervalle $]lh,ph[$. Comme ces suites sont positives par définition (probabilités), nous venons donc de démontrer le premier cas de figure p>l (l'existence d'un h_{min} unique tel que défini dans la proposition initiale est triviale : c'est la borne inférieure des solutions (h,i(h)).

Dans le cas de figure l=p, il suffit de reprendre les relations de récurrence concernant $(u_{h,i})$ et $(u'_{h,i})$ pour constater directement que $u_{h,i}u'_{h,i} = 1$ pour tout h et tout $i \leq h$. Par conséquent, $u_{h,i}$ et $u'_{h,i}$ ne peuvent pas être simultanément inférieurs à 0.5. Donc, pour tout $\epsilon < 0.5$, il n'existe aucun couple de solutions (h,i) vérifiant les deux hypothèses.

Considérons à présent le dernier cas de figure (l>p). Soit $(S_{h,i})$ la suite définie par :

$$\forall h \in \mathbb{N}, \forall i \in \mathbb{N}, i \leq h,\ S_{h,i} = u_{h,i} + u'_{h,i}$$

Nous allons montrer par récurrence sur h que chaque terme $S_{h,i}$ est supérieur ou égal à 1.
Pour h=1, $S_{1,0} = 1$ et $S_{1,1} = (1-p)+l > 1$. L'hypothèse de récurrence est donc vraie pour h=1.
Supposons que cette hypothèse soit vraie jusqu'au rang $h \geq 1$. Posons $l = p + \alpha$, avec $\alpha > 0$.
Nous allons déterminer une relation de récurrence liant les $S_{h,i}$. Pour cela, nous utilisons celle qui existe entre les $u_{h,i}$ et entre les $u'_{h,i}$, qui permet d'écrire :

$$u_{h+1,i+1} + u'_{h+1,i+1} = (1-p)u_{h,i} + p.u_{h,i+1} + (1-p-\alpha)u'_{h,i} + (p+\alpha)u'_{h,i+1}$$

En regroupant les termes, il vient :

$$S_{h+1,i+1} = (1-p)S_{h,i} + p.S_{h,i+1} + \alpha(u'_{h,i+1} - u'_{h,i})$$

Or, $u'_{h,i+1} - u'_{h,i} = v'_{h,i+1} \geq 0$. Par conséquent, on a l'inégalité suivante :

$$S_{h+1,i+1} \geq (1-p)S_{h,i} + p.S_{h,i+1}$$

Par hypothèse de récurrence, $S_{h,i} \geq 1$ et $S_{h,i+1} \geq 1$. On en déduit que $(1-p)S_{h,i} + p.S_{h,i+1} \geq 1$ (relation barycentrique). D'où $S_{h+1,i+1} \geq 1$. Cela étant vrai pour tout $i \in \{0,...,h\}$, et sachant que $S_{h+1,0} = 1 \geq 1$, on en déduit que $\forall i \in \{0,..,h+1\}$, $S_{h+1,i} \geq 1$. Ce qui termine la récurrence. Nous venons donc de prouver le cas l>p. En effet, la somme de deux termes étant supérieure ou égale à 1, l'un des deux termes doit être supérieur ou égal à 0.5 .

◀

Troisième partie

A parte : Réflexion informelle autour de notre modélisation

5

Plausibilité de notre modèle au regard du vivant

Connais-toi toi-même et tu connaîtras l'univers et les dieux

Inscription sur le fronton du temple de Delphes

5.1 Idées directrices

Au début de notre avant-propos, nous avons souligné les réussites de certains modèles inspirés du vivant, en robotique mobile. Nous avons noté que, malheureusement, ces modèles ne sont pas compatibles avec notre postulat, car il manque une grande partie de formalisation.

À présent que nous avons constitué notre modélisation d'un système apprenant des actions réflexes, il nous semble intéressant de regarder dans quelle mesure il est plausible d'un point de vue du vivant.

Bien entendu, cette plausibilité ne s'applique pas à un niveau physique (notre système est abstrait), mais au niveau des concepts utilisés.

Notre modélisation est fondée sur l'existence de **contraintes internes** à un système, qui doivent être respectées à chaque instant. D'un point de vue biologique, cela est à rapprocher de *constance du milieu intérieur* de Claude Bernard et d'*homéostasie* de Walter Cannon [Cannon, 1932].

Toutefois, **notre notion de contrainte va au delà d'un mécanisme physiologique de contrôle** : dans notre modélisation, **les contraintes ne sont pas directement observables**, mais leur existence implique que le système auquel celles-ci s'appliquent possède certaines propriétés observables, qui sont **des conséquences** de l'application de contraintes au système, mais

aussi de l'interaction de ce dernier avec son environnement.

Dans cette optique, nous nous intéresserons plus particulièrement à **l'apprentissage perceptif**, dont le terme est emprunté à Gibson. La démarche propre à notre modélisation suppose que l'ensemble des informations perceptives est le résultat d'une structuration interne à l'entité apprenante, dont le résultat peut s'expliquer en utilisant deux postulats :

1. la particularité des informations perceptives est due à l'interaction avec l'environnement dans lequel l'entité est **réellement** plongé
2. la structure d'un ensemble d'informations perceptives obéit aux contraintes d'observabilité et d'unicité, **quel que soit l'environnement**

Voici les conséquences de notre modèle sur la nature de l'information perceptive :

1. l'information perceptive est le résultat d'**un processus d'anticipation** (conséquence direct de notre construction du sous-système d'apprentissage perceptif, qui est un détecteur)
2. l'information perceptive est **obtenue au bout d'une certaine durée** (conséquence de l'utilisation d'un mécanisme de sélection)
3. l'information perceptive est **multi-modale** (conséquence de la contrainte d'unicité sur des événements simultanément détectés, mais étant associés à des grandeurs physiques distinctes)
4. l'information perceptive est liée avec les notions de **régularité** et de **redondance** (conséquences de la contrainte d'observabilité)
5. la détection d'une information perceptive est indissociable du **sentiment de certitude** concernant la validité de cette détection (l'information perceptive est basée sur la notion de fiabilité qui sont liées aux caractères de régularité et de redondance)

La quatre premières caractéristiques sont ou ont été l'objet de travaux dans le domaine des neurosciences, de la psycho-physique ou de la psychologie. L'influence de l'environnement (contexte) sur la perception et la limite de cette influence sont également discutées. Par contre, le dernier point est plus surprenant.

L'objectif de ce chapitre est de donner **des éléments de justification** à chacun de ces points, dont il faut se souvenir qu'ils sont des propriétés secondaires, mais observables, de notre modélisation. En ce sens, celle-ci induit leur existence. Donc, ce chapitre constitue **l'ébauche d'une réflexion qui appuierait ou rejetterait la possibilité de transposer notre modèle abstrait dans le monde réel**.

5.2 Caractéristiques de l'apprentissage perceptif

5.2.1 Différenciation et unification

Le terme *apprentissage perceptif* que nous avons choisi est-il cohérent avec celui qui est utilisé dans les domaines du vivant ?

Les capacités de perception se forment à partir de l'expérience : il existe bien un *apprentissage perceptif*. Ce terme, mis en valeur par l'*approche écologique* de Eleanor Gibson [1] , regroupe un ensemble de mécanismes, que nous ne détaillerons pas dans ce document [2]. Nous en soulignons

1. On pourra consulter [Gibson et Gibson, 1955] et [Gibson, 1969]. 2. Le lecteur pourra se reporter à [Goldstone, 1998].

deux ici : les mécanismes de **différenciation** et d'**unification**.

Le premier mécanisme suppose qu'il n'y a pas ajout de connaissances au niveau perceptif, mais qu'on part avec une capacité de discrimination réduite, qui se traduit par un faible nombre de catégories perceptives, et que l'expérience permet de transformer ces catégories grossières. Gibson a montré que ce phénomène pouvait se produire sans utilisation de mécanismes de conditionnement ou autres d'autres retours d'information [Gibson et Gibson, 1955]. **C'est dans ce sens que nous séparons l'AO de l'AP dans notre modèle**.

Pour illustrer ce qu'est le mécanisme de différenciation, on peut citer des exemples pour lesquels il est communément reconnu qu'un sens s'éduque avec l'expérience : musique ou dégustation de vin. Les mélomanes savent très bien que l'oreille s'éduque et qu'on peut ne « comprendre » une musique qu'après un certain nombre d'écoutes, sans qu'aucune faculté « intellectuelle » ne soit mise en jeu. On peut citer également l'apprentissage d'une langue étrangère, qui met en œuvre la construction d'une capacité d'écoute, propre à cette langue, permettant de déchiffrer la « mélodie » de celle-ci. D'autre part, des expériences menées sur des nourrissons montrent qu'ils reconnaissent d'abord le timbre de la voix de leur mère, puis la mélodie de sa langue maternelle.

Le terme de différenciation peut également s'appliquer à la formation de la vision. Des études montrent que le bébé possède une acuité visuelle très faible à la naissance et que celle-ci se développe au cours des premières semaines [Fantz et al., 1962]. Cela ne signifie pas que ses organes de perception ne sont pas formés, mais que sa structuration neuronale, qui donne la capacité à percevoir, n'est pas achevée [Teller, 1981].

Le mécanisme d'unification, au contraire du précédent, permet de regrouper des *stimuli* qui étaient perçus comme étant différents en un tout unique. La reconnaissance de visage est un exemple de ce mécanisme : il a été suggéré que le visage est analysé comme un tout et non comme une somme de spécificités [Farah, 1992]. L'identification d'un mot en un tout en est également un exemple.

Dans [Goldstone, 1998], Goldstone déclare que les mécanismes de différenciation et d'unification ne sont pas contradictoires, paradoxalement : les phénomènes ayant lieu **simultanément** ont tendance a être regroupés en un seul, alors que les sources d'informations qui **ne sont pas corrélées** auraient tendance à être séparées.

L'ensemble de ces faits semble être en cohérence avec notre notion d'apprentissage perceptif. La contrainte d'unicité règle la formation des états, selon le principe de simultanéité, et engendre les phénomènes de différenciation et d'unification.

5.2.2 Influence du contexte sur le mécanisme de différenciation

Notre modélisation implique que l'interaction avec l'environnement détermine l'ensemble des perceptions discriminables. Cela signifie que la spécificité de l'environnement influence directement la nature de cet ensemble. Dans notre approche, la deuxième partie de la contrainte d'observabilité oblige le système à engendrer une information perceptive à partir de signaux émis en réalité. Nous avons proposé, en perspectives, un mécanisme permettant de modifier progressivement un ensemble d'événements de manière à ce que ceux-ci coïncident avec ceux détectés en réalité. Nous avons supposé que les génératrices des focus pourraient être modifiées dans cette

opération. Dans ce cas, et sous la contrainte que la mesure de l'ensemble d'événements reste plus petite qu'un certain ϵ fixé *a priori*, notre mécanisme aboutit au fait que seules les hypothèses pouvant être réellement validées existent dans la mémoire, au bout d'un certain temps.

Des exemples semblent aller dans le sens de l'importance du contexte perceptif. Ainsi, « les inuits possèdent dans leur langage une cinquantaine de mots pour identifier les différentes formes que prennent l'eau, la neige et la glace »[1], ce qui signifie qu'ils sont eux-mêmes capables de différencier ces nuances (focalisation des aptitudes visuelles autour d'une gamme très étroite de signaux perceptifs).

On peut également restreindre expérimentalement l'étendue du contexte perceptif et en observer les conséquences : des animaux dont les yeux sont exposés continuellement à une seule couleur dès la naissance deviennent incapables de discerner les couleurs. Le résultat de l'expérience exposant continuellement une souris à des barres verticales aboutit, si elle est poursuivie assez longtemps, à une perte de la capacité de percevoir des barres horizontales. Dans le même ordre d'idées, si on interdit la possibilité d'apprentissage de la pince fine[2] à un enfant pendant trop longtemps, ce geste deviendra impossible à effectuer par l'adulte.

L'aspect temporel de la formation de la perception semble très important. Des expériences sur la discrimination de phonèmes n'existant pas dans la langue maternelle du bébé font apparaître que le bébé âgé de moins de onze mois discrimine immédiatement un phonème de sa langue et un autre qui n'y est pas inclus, mais qui est phonétiquement proche. Au delà de onze mois, cette capacité disparaît (d'après [Werker et Lalonde, 1988]).

Nous remarquons ici une limite de notre modèle, qui n'explique pas cette contrainte temporelle.

5.2.3 Utilisation de contraintes

Notre modélisation impose qu'il existe à la fois une interaction entre le système et son environnement, mais également un ensemble de contraintes internes à ce système. L'information perceptive est le résultat de ce double processus.

Il faut souligner le rôle particulier que possèdent chacun des deux mécanismes. Comme nous l'avons dit dans le paragraphe précédent, l'interaction avec l'environnement permet de construire la spécificité des informations perceptives. Mais, l'existence de contraintes sur le système est la **la cause des modifications de celui-ci**, ce qui engendre l'apprentissage.

Dans notre cas, les contraintes permettent de réduire l'ensemble des mémoires valides, donc elles diminuent également l'ensemble des évolutions possibles d'une mémoire au cours du temps. **Cette diminution doit être rapprochée de celle produite par l'ajout de connaissances** *a priori* **dans un système artificiel apprenant**.

Il est montré que les systèmes vivants utilisent des contraintes, qui leur permettent de mieux s'adapter à leur environnement [Goldstone, 1998]. Ainsi, Eimas montre que les bébés possèdent

1. op. cit. Pierre Dansereau 2. Il s'agit de la capacité humaine à rapprocher le pouce et l'index pour saisir un objet finement et non à pleine main.

à la naissance des techniques de segmentation de la parole, qui leur servent par la suite à comprendre la signification des phrases [Eimas et al., 1971]. L'utilisation de contraintes est également nécessaire dans le domaine de la vision, pour rendre la perception non ambiguë. La détermination du flot optique est un bon exemple de la nécessité d'utiliser des contraintes, sans quoi le problème est mal posé [Barron et al., 1994] (*problème d'ouverture* entre autres). Ces contraintes sont des pré-supposés sur la manière la plus adéquate de traiter l'information visuelle à un instant donné. Elles n'ont aucun rapport avec les contraintes de notre modélisation.

Nos contraintes agissent dans la création d'un ensemble d'informations perceptives valide (respectant CO et CU). Cependant, il semble réaliste de penser qu'il en existe plus d'un. Quel ensemble utiliser à un instant donné ? Notre modélisation ne donne pas de réponse à cela, actuellement. C'est à ce niveau que des contraintes similaires à celles décrites dans le problème de la vision seraient nécessaires.

La contrainte d'observabilité de notre modélisation oblige à considérer des ensembles d'événements associés avec une certaine **régularité** ou une certaine **redondance** de l'environnement. Cela doit être rapproché de la tentative faite par H. Barlow d'associer des configurations neuronales ayant une faible probabilité de se produire avec une quantité d'informations élevée [Barlow, 1985], en suivant la démarche de Shannon. À ce propos, le lecteur pourra lire avec intérêt [Barlow, 2001a] et [Barlow, 2001b]. Celui-ci suggère que la redondance ne peut pas être expliquer à la manière de la théorie de l'information de Shannon, c'est-à-dire par rapport à la capacité d'un canal de transmission.

5.3 Caractéristiques de l'information perceptive

5.3.1 Notion d'information perceptive

Notre notion d'information perceptive possède des caractéristiques particulières :

1. l'espace dans lequel évolue l'information perceptive est **un espace événementiel**, qui n'est pas directement connecté avec les grandeurs physiques
2. cet espace est muni d'une mesure de simultanéité (on suppose qu'on est capable de savoir si deux événements sont détectés simultanément)
3. l'information perceptive est une conjonction d'événements pouvant être théoriquement **simultanés**
4. l'information perceptive n'est pas une mesure au sens du traitement du signal, mais plutôt **une mesure au sens de la mécanique quantique** (voir le chapitre suivant)
5. l'information perceptive est associée à notre notion de fiabilité

Dans notre modélisation, il faut bien comprendre que ce qui peut être mesuré est, **par définition de notre mesure**, l'information perceptive. Ainsi, les événements élémentaires la composant ne peuvent pas être mesurés en règle générale et ne forment donc pas des informations perceptives.

Chaque événement élémentaire est détecté par un mécanisme qui met en jeu une ou des grandeurs physiques (dans notre modélisation, l'élément physique est le *focus*). Les événements composant une information perceptive pouvant être liés à des grandeurs physiques différentes, **l'information perceptive est multi-modale**. Nous supposons ainsi qu'**il est impossible de**

mesurer séparément (donc avec fiabilité) ces différentes composantes.

Le mécanisme de détermination de l'information perceptive nécessitant une certaine durée (liée au paramètre h), nous supposons également qu'aucune mesure ne peut être effectuée en deçà de cette durée (avec fiabilité).

5.3.2 Postulat d'existence de l'information perceptive

Notre définition de l'information perceptive semble très abstraite. Pourtant, **nous supposons son existence et son rôle prépondérant dans les mécanismes intimes du vivant**. Voici notre postulat :

Postulat 2 *Les informations perceptives constituent les briques qui sont utilisées par le savoir procédural. La détection d'une information perceptive est interprétable en tant que* **mécanisme d'anticipation**. *De plus, cette détection est associée avec* **le sentiment de certitude** *sur la réalité de sa propre existence, sans pour autant pouvoir que l'individu percevant puisse prouver (d'une manière procédurale) cette existence. Nous appelons cela* **le paradoxe de l'évidence**.

Une conséquence très importante de cela est que nous séparons les notions de connaissance (ou de savoir) supposée vraie de celle d'information perceptive, qui s'impose à l'esprit comme étant la réalité. Ainsi, nous supposons que l'information perceptive est ce qui est ressenti comme étant vécu, à la différence de la connaissance, qui est un assemblage d'informations perceptives.

L'apprentissage d'une connaissance, de quelque nature qu'elle soit, nécessite donc un apprentissage procédural (que nous avons appelé AO), qui utilise des informations perceptives supposées être apprises grâce à l'AP.

L'objectif des paragraphes qui suivent est d'expliquer notre position. Ils sont constitués d'un ensemble d'idées qui sont associées à notre postulat.

5.3.3 L'information perceptive est soumise au paradoxe de l'évidence

> *Tant que tu parles autant je souffre comme un damné malheureusement tu vois tout de travers tu n'as aucune idée d'une vie dans un isolement absolu c'est plus grave qu'un emprisonnement ou qu'une prétendue détention je me noie dans la solitude*
>
> Birger Sellin, autiste[1]

Le rôle de l'AP est de donner naissance à la capacité de produire de *l'information perceptive* à partir d'un ensemble de signaux. Mais cette capacité va plus loin que l'utilisation des cinq sens. Ainsi, un bon joueur d'échecs « verra » l'échiquier d'une autre façon qu'un débutant. Le

1. extrait de « Une âme prisonnière », édition Robert Lafont, 1994

problème de ce dernier (et ce n'est pas difficile de l'expérimenter soi-même) est qu'il ne voit rien justement : la disposition des pièces n'a pas de sens pour lui. Dans ce cas, il est étonnant de constater que le débutant va procéder un peu comme un ordinateur, par essai mental de tous les coups possibles ou d'un éventail de coups pris un peu au hasard. Par contre, contrairement à l'ordinateur, l'homme ne possède qu'une très petite « puissance de calcul », d'où des résultats modestes lorsqu'on ne possède pas une certaine *perception* du jeu. Par contre, le très bon joueur va écarter d'emblée une très grande majorité des coups possibles en ne gardant que ceux qui sont intéressants : il possède une grande capacité à **filtrer l'information**, c'est-à-dire à aller chercher l'information pertinente.

Cependant, l'information perceptive possède une caractéristique qui lui est propre : il est impossible d'expliquer (sans rentrer dans des considérations biologiques d'étude des organes impliqués) la manière dont on perçoit, et encore moins le cheminement qui nous a permis d'établir notre faculté de perception : essayez d'expliquer à un aveugle ce qu'est une couleur sans utiliser un vocable visuel. Cela ne signifie pas que celui-ci est incapable de percevoir, d'une autre façon, ce genre de phénomène. Mais cette faculté, qui est liée à une intime conviction, est difficilement communicable ou enseignable.

Nous nous trouvons donc face à un paradoxe : **l'AP crée une source d'informations qui s'imposent à nous comme étant évidentes mais nous sommes incapables d'expliquer la nature de cette information ni, par conséquent, de la transmettre à autrui**. Au contraire, lorsque cette source d'informations n'existe pas, nous n'avons aucune possibilité pour l'appréhender et la comprendre (autrement que par le biais d'une étude scientifique du phénomène). L'AP est si intime qu'il s'établit une frontière à la fois entre l'avant et l'après apprentissage (on ne peut très rapidement plus s'imaginer comment on voyait le problème avant) et, parallèlement, entre celui qui a appris et celui qui n'a pas appris. L'existence de cette barrière est décrite très concrètement dans l'essai intitulé *Flatland*[1] [Abott, 1952].

Toute personne ayant tenté d'apprendre un sport a fait l'expérience de ce paradoxe : il ne suffit pas de répéter inlassablement un geste ou un ensemble de gestes pour qu'ils soient mémorisés et reproduits parfaitement le moment voulu ; cela appartient au domaine de l'acquisition de connaissances. Les bases nécessaires se trouvent dans la sensation (indescriptible) que le geste effectué est correct ou non. Celle-ci sert de repère à l'auto-évaluation du niveau de performance auquel on est arrivé, ce qui permet une progression. Elle passe par la possibilité de créer une représentation mentale précise du geste à effectuer. Sans l'aide de ce repère, les mêmes erreurs sont commises régulièrement, sans pouvoir vraiment les corriger. Pour un sportif de haut niveau, il semble que la faculté de représentation mentale du geste idoine (incluant la mobilisation des parties du corps concernées) est très développée ; cependant, le *stress* lié à l'importance de l'événement sportif peut réduire l'acuité de cette perception.

Mais, ce paradoxe existe aussi dans l'apprentissage de connaissances plus scolaires, et apparemment abstraites : le fait de ne rien comprendre se traduit par l'absence de création d'images mentales chez l'étudiant, ce qui signifie que le discours ne lui parle pas. La faculté d'abstraction

1. *Flatland* est raconté par un être imaginaire vivant dans un monde possédant deux dimensions. Faisant en rêve la rencontre avec un être issu d'un monde mono-dimensionnel, il tente en vain de faire comprendre à celui-ci la possibilité d'exister en deux dimensions. Cette histoire montre très simplement que la possibilité d'appréhender le monde dépend avant tout de ses propres sens.

n'est pas la possibilité de manipuler des symboles non signifiants : c'est au contraire la faculté à donner un sens à ces objets, pour pouvoir ensuite les utiliser.

Avant tout, **le paradoxe de l'évidence pose le problème de l'apprentissage de la faculté de percevoir** : celle-ci n'est pas transmissible dans une relation maître/élève et est assimilée d'une manière non volontaire. En particulier, nous pensons que la réussite du processus d'imitation, qui met en jeu un objectif précis, nécessite au préalable un très bon fonctionnement des facultés de perception nécessaires à l'atteinte de l'objectif[1]. Par conséquent, la faculté d'imitation est hiérarchiquement dépendante des capacités perceptives, qui sont soumises au paradoxe de l'évidence : la cause de l'échec d'un apprentissage d'objectif est, à notre avis, autant liée à un manque du processus de perception qu'à une mauvaise utilisation de ce processus pour atteindre l'objectif.

5.3.4 Rôle des signaux perceptifs internes

Jusqu'à présent, nous avons évoqué des exemples utilisant des signaux perceptifs externes (à travers la vision ou l'ouïe). À travers ces exemples, nous avons supposé que l'AP n'est pas guidée par un objectif précis (apprentissage latent). Dans notre exposé, l'AO utilise des signaux internes (signaux de renforcement) pour marquer les états du système, c'est-à-dire les différentes catégories d'informations perceptives. Comment peut-on justifier une telle organisation ? C'est l'objectif de cette sous-section.

Il nous faut tout d'abord préciser le terme *signal interne*. Il s'agit d'un message (chimique ou électrique) qui est émis par différentes parties du corps qui pourra être utilisé en tant que *source d'information interne*. Le signal de renforcement en est un. Mais les émotions, en général, suivent également cette définition. Par « émotion », il faut comprendre un champ plus vaste que les émotions primaires (joie, peur, colère, etc...). D'une manière plus générale, ce sont des signaux que certaines parties du corps génèrent, et qui sont perçus d'une manière consciente ou non par l'individu. Des expériences menées avec des personnes dont la moelle épinière a été sectionnée, rendant insensible une plus ou moins grande partie de leur corps (suivant l'endroit de la section) montrent que celles-ci souffrent de déficiences partielles au niveau du ressenti des émotions ; ces problèmes sont d'autant plus accentués que la zone insensible du corps est importante.

Les travaux entrepris par Damasio [Damasio, 1994] tendent à montrer que les émotions jouent un rôle important dans des processus intellectuels rationnels. Cette idée a été redécouverte récemment, alors qu'elle avait été émise il y a longtemps par James ou Darwin, puis mise à l'écart. Il est vrai que la relation entre émotion et rationalité n'est pas évidente *a priori* ; il paraît même aller de soi qu'une réflexion rationnelle (objective) doit exclure toute émotion (subjective, par essence). Pourtant, des études menées sur des patients souffrant de lésions dans des aires précises de la région préfrontale du cerveau montrent cette relation. En effet, il semble que ces personnes sont à la fois incapables d'éprouver la moindre émotion et, parallèlement, ont un trouble prononcé de la capacité à décider de façon avantageuse dans leur vie quotidienne. Ainsi, bien que ces hommes aient conservé des capacités intellectuelles intactes et utilisent encore tous les instruments de la rationalité, ils prennent dans de nombreux cas des décisions irrationnelles, très souvent à leur désavantage. À partir de cette constatation, Damasio formule une hypothèse selon laquelle le mécanisme du raisonnement reçoit des signaux internes, consciemment ou non, de la

1. Par exemple, il a été prouvé que les enfants dont on n'avait pas détecté un défaut de vision avaient plus de mal à apprendre à lire

part des centres émotionnels, permettant au raisonnement d'aboutir ; dans les cas pathologiques, ces signaux ne sont plus reçus ou interprétés : ils ne sont plus transformés en information perceptive.

Cette hypothèse a été baptisée « hypothèse des marqueurs somatiques ». Les marqueurs somatiques joueraient le rôle de filtres permettant, au cours d'un raisonnement quelconque, de supprimer d'office des solutions qualifiées de « mauvaises ». Il ne resterait alors plus que quelques options qui seraient accessibles à la conscience et à partir desquelles l'individu pourrait choisir au mieux.

Le choix d'un ensemble fini de marquages possibles dans notre algorithme CbL est inspiré de l'hypothèse des marqueurs somatiques. L'algorithme de choix de la commande fait apparaître l'ensemble des commandes ayant un marquage maximal, qui sont jugées comme étant de qualité comparable. Notre notion d'optimalité est donc affaiblie par rapport à celle utilisant une mesure continue (les marquages sont choisis dans un ensemble fini).

5.3.5 Généralisation de la notion de perception aux signaux internes accompagnant le mouvement

Nous suivons une hypothèse actuellement discutée en psychophysique et en psychologie expérimentale, que l'action participe activement à la genèse de la perception.

Tout d'abord, il faut bien différencier la sensation du mouvement et les signaux moteurs qui commandent la réalisation du mouvement. Il s'agit d'un signal qui accompagne le mouvement et passe inaperçu dans des actes routiniers. Pourtant, il est d'une très grande importance, car il fournit une information sur l'exécution effective du mouvement. Les personnes handicapées disposant d'une prothèse à la place d'un membre peuvent éprouver des difficultés par le manque de retour informatif. Au contraire, un danseur de très bon niveau est capable de connaître précisément la position des différentes parties de son corps grâce à cette information : cela lui permet d'exécuter le mouvement juste. Le sportif de haut niveau utilise abondamment ce retour informatif : l'exécution d'un mouvement complexe sur un trampoline nécessite plus d'utiliser ces informations que l'information visuelle (même si celle-ci est indispensable également).

Notre hypothèse stipule que la sensation du mouvement peut être développée par apprentissage, de manière à être utilisée pour créer une source d'informations perceptives, de la même manière que les signaux externes. Lorsque nous parlerons d'actions réflexes dans la section suivante, nous supposerons que celles-ci sont accompagnées de la production de signaux perceptifs particuliers à l'organe moteur.

Il semble que la perception se fasse en utilisant des actions réflexes. Ainsi, des études menées concernant la reconnaissance de scènes ou, plus particulièrement, de visages, ont montré l'importance du phénomène des saccades oculaires. O'Regan formule l'hypothèse que cette action interne a pour but d'aller chercher l'information dans la scène visuelle, là où l'individu pense qu'elle doit se trouver. Cela signifie qu'en dehors des zones explorées activement par l'œil, des changements de la scène ne sont par perçus. Des expériences de psychologie expérimentale allant dans ce sens sont proposées dans [O'Regan et al., 1999] ou [O'Regan et Noé, 2001]. L'idée d'O'Regan est que l'information visuelle est le résultat d'un processus dynamique (impliquant l'utilisation d'actions réflexes) stimulé par la nécessité d'aller chercher l'information. Cela signi-

fie que nous supposons que le processus de perception est essentiellement dynamique et que la reconnaissance d'une situation perceptive ne peut pas être effectuée instantanément, mais après l'exécution d'une certaine séquence d'actions internes à l'individu. Notre schéma dynamique du processus de sélection est en accord avec cette idée.

5.3.6 Mécanisme d'anticipation

Nous pensons que notre modèle de détecteur peut être interprété comme un *mécanisme d'anticipation* permettant d'aboutir à l'information perceptive. Le phénomène d'anticipation est actuellement au cœur de débats scientifiques sur l'existence ou non d'un *modèle interne* de l'environnement.

Le lancé de balle est un bon exemple pour illustrer ce qu'est l'anticipation : lorsqu'un enfant joue pour la première fois avec une balle en la lançant verticalement, il effectue un mouvement vers la balle pour la rattraper lorsqu'elle redescend, provoquant un choc important ; mais un réflexe se met en place, par apprentissage, pour « amortir » ce choc. Il a été montré que des signaux moteurs sont transmis à la main quelques millisecondes avant l'impact de la balle, de manière à faire descendre la main en anticipation du choc. Une hypothèse serait que l'individu utilise son expérience de manière inconsciente pour effectuer ce mouvement (voir [Hanneton et al., 1998]).

Une autre expérience montre une caractéristique de ce processus d'anticipation. On lâche une balle avec une certaine vitesse à la verticale d'un cosmonaute. Bien que celui-ci sache que la balle n'accélère pas, les signaux moteurs au niveau de sa main sont enregistrés au moment même où la main aurait dû descendre si les conditions de pesanteur avaient été présentes. Mais dans un cas d'apesanteur, ces signaux se déclenchent trop tôt.

Notre modélisation traduit cette faculté d'anticipation en posant une mémoire comme centre de ce mécanisme. Or, la mémoire étant structurée grâce à la perception et aux actions réflexes, l'anticipation se comporte comme un système prédictif sur l'évolution de la perception, couplé **simultanément** à l'envoi d'un signal moteur. Par cela, nous écartons le schéma classique impliquant une relation de causalité entre la perception et l'action (on perçoit, puis on agit en réponse de cette perception).

D'autre part, nous pensons que la faculté d'anticipation est une réponse à l'incertitude. En effet, dans notre modèle, un état perceptif peut regrouper des événements qui ne sont pas associés à des valeurs connexes de grandeurs physiques. En d'autres termes, **notre mécanisme n'a pas pour objectif de prévoir exactement ce qui va arriver** : ce serait d'une part objectivement irréaliste et, d'autre part, cela nierait le caractère, non prévisible par essence, de l'interaction entre l'entité et le monde extérieur. Le processus d'anticipation permet d'établir **un ensemble de bifurcations possibles**. Par conséquent, à un instant donné, on possède un ensemble de *scenarii* d'évolution. Et nous supposons qu'à un moment donné on connaîtra effectivement le *scenario* qui se déroule effectivement.

Pour illustrer ce discours, l'étude du comportement des hommes au sein des marchés financiers, évoquée par Sauvage[1], est révélatrice [Sauvage, 1999]. Ce domaine est objectivement l'un de ceux où le risque et l'incertitude sont les plus évidents. L'utilisation simple des probabilités

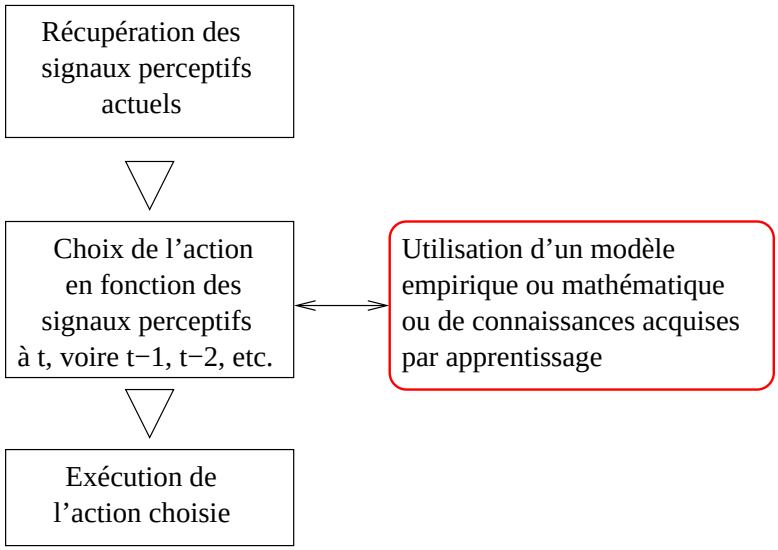
1. Gérard Sauvage est un économiste spécialisé dans la prévision macro-économique et financière.

voudrait que sur une longue période de temps, les résultats d'investisseurs « avertis » soient très proches les uns des autres. Or, on trouve deux contre-exemples célèbres à cela : Sorros et Buffet, ayant réalisé des performances nettement au dessus de la moyenne sur les 30 dernières années [1]. Pourtant, leurs stratégies respectives sont totalement opposées : Sorros effectue des placements sur des durées inférieures au mois, alors que Buffet a une vision basée sur le long terme (durée de placement supérieure à 5 ans). Cependant, ils ont en commun une capacité étonnante à contourner l'incertitude, chacun à sa façon. Ainsi, Sorros utilise pleinement le schéma d'établissement de *scenarii* tel que nous le présentons dans ce recueil : Sauvage dit : « Pour Georges Sorros, l'évolution du marché ne résulte pas d'une succession de faits ou d'informations, mais de l'existence d'une dynamique sous-jacente [...]. Raisonner en termes de dynamique, c'est déterminer une trajectoire, d'où découle la capacité de faire des prédictions à très court terme sur l'évolution des cours ; mais c'est aussi savoir qu'à tout moment, cette trajectoire peut être déviée pour une raison ou pour une autre, ce qui le conduit à une vigilance permanente afin de détecter le plus vite possible les changements d'une dynamique. ». La capacité d'anticipation de Sorros lui permet de créer des schémas d'évolution possibles, puis de détecter au plus vite celui qui est en train de se produire effectivement. Il faut noter que cette capacité n'est pas d'ordre « intellectuel » : Sorros n'utilise aucune théorie économique, mais, comme il le déclare, agit avec du *filling* : « Je réagis aux événements sur les marchés comme un animal réagit aux événements dans la jungle ». Buffet, quant à lui, se prémunit des fluctuations « locales » du marché, en effectuant des placements long terme sur des sociétés dont il anticipe le potentiel, tant objectif que subjectif (nature et cohésion de l'équipe dirigeante, par exemple).

L'existence de l'anticipation est essentielle pour comprendre le raisonnement qui a été développé dans la deuxième partie de ce document. Elle change radicalement la manière d'aborder la résolution d'un problème d'interaction perception/action. En effet, dans le cas « classique », on suppose qu'au moment présent (disons l'instant t), la réponse de l'entité aux données perceptives reçues à t (voire à t-1, t-2, etc.) est une fonction de ces dernières. Cela signifie concrètement que la réponse de l'entité par rapport à ces données perceptives ne peut être connue qu'à partir de l'instant t, et qu'elle dépend très fortement de celles-ci (lien fonctionnel). Cette réponse découlera des connaissances - *a priori* ou acquises par apprentissage - que possède l'entité à l'instant t, celles-ci définissant intégralement le lien fonctionnel entre les entrées du système (données perceptives) et les sorties (réaction de l'entité). Ce processus est schématisé par la figure 5.1. On voit bien que l'utilisation de la connaissance qu'on a sur le problème intervient au moment du choix de l'action, après l'instant t. Donc, ce choix est donné *a posteriori*, en retard par rapport aux perceptions qui l'ont engendré. Si on transpose ce processus au niveau comportemental humain, cela signifie que lorsqu'on accomplit une tâche, même routinière, il existerait un choix, conscient ou non, à chaque pas de temps, déterminant l'action adéquate à cet instant précis. En outre, cela implique que la tâche n'est pas perçue dans sa durée et sa globalité, et qu'on est incapable d'utiliser l'expérience passée pour prévoir la suite de la séquence d'actions à partir de cet instant t.

Au contraire, nous pensons que l'entité apprenante établit une représentation mentale du problème, composée d'un ensemble de *scenarii*, qui sont autant d'hypothèses d'évolution possibles du problème. Celles-ci sont des fragments de la représentation mentale globale du problème ; donc, elles ont été construites à partir de signaux perceptifs et émotionnels, ainsi qu'à partir d'actions internes et externes. Les signaux perceptifs sont utilisés au moment présent pour valider ou invalider chacune de ces hypothèses. Ainsi, la réalité de l'instant présent apparaît à

1. Plus de 30% par an sur 30 ans, alors que la moyenne est inférieure à 10%



À l'instant t, le système sélectionne l'action à exécuter en fonction de sa perception présente, voire de ses perceptions passées. Ce lien fonctionnel est exprimé, soit grâce à un modèle du problème auquel le système est confronté, soit à l'aide des connaissances acquises par apprentissage. Le principal problème d'un système réactif comme celui-ci est la dépendance forte de l'action à entreprendre par rapport aux perceptions reçues à l'instant t : pour que l'action ne soit pas erronée, cela suppose qu'on soit capable de traiter le signal reçu de manière à ce qu'il corresponde à une réalité objective (réalité induite par le modèle qu'on utilise ou par les connaissances acquises).

Fig. 5.1 – Schéma classique du traitement du phénomène d'association perception/action

l'entité comme l'ensemble des hypothèses d'évolution « plausibles » du problème, c'est-à-dire qu'une partie limitée de la représentation mentale du problème est « active ». Ce point précis mérite d'être approfondi.

Nous pensons qu'une hypothèse (ou un *scenario*) ne peut être validée ou invalidée instantanément : les données perceptives, prises dans le temps, vont être utilisées pour renforcer ou diminuer le *sentiment* de justesse de l'hypothèse, pour aboutir à un moment donné à une certitude la concernant. Et c'est la sensation d'être certain qu'une hypothèse est correcte ou fausse qui conditionne la validation ou l'invalidation du *scenario*. La figure 5.2 montre que le processus que nous décrivons ne correspond pas au schéma décisionnel « classique » de la figure 5.1 : on ne se préoccupe pas véritablement du choix d'une action précise à l'instant t, puisque celui-ci est exprimé dans les *scenarii* d'évolution possible du complexe perception/action. La tâche sera considérée comme correcte si les hypothèses sont régulièrement validées aux cours du temps. Ainsi, nous admettons que le besoin sécuritaire se traduit, après apprentissage, par cette sensation de certitude qui doit accompagner la résolution du problème à chaque pas de temps. Ainsi, d'un point de vue interne à l'entité, la tâche à accomplir se décompose en une succession d'hypothèses « actives ». D'autre part, on peut remarquer que le mécanisme de validation ou d'invalidation ne fait appel à aucun choix de la part de l'entité et que ce processus est en grande partie inconscient.

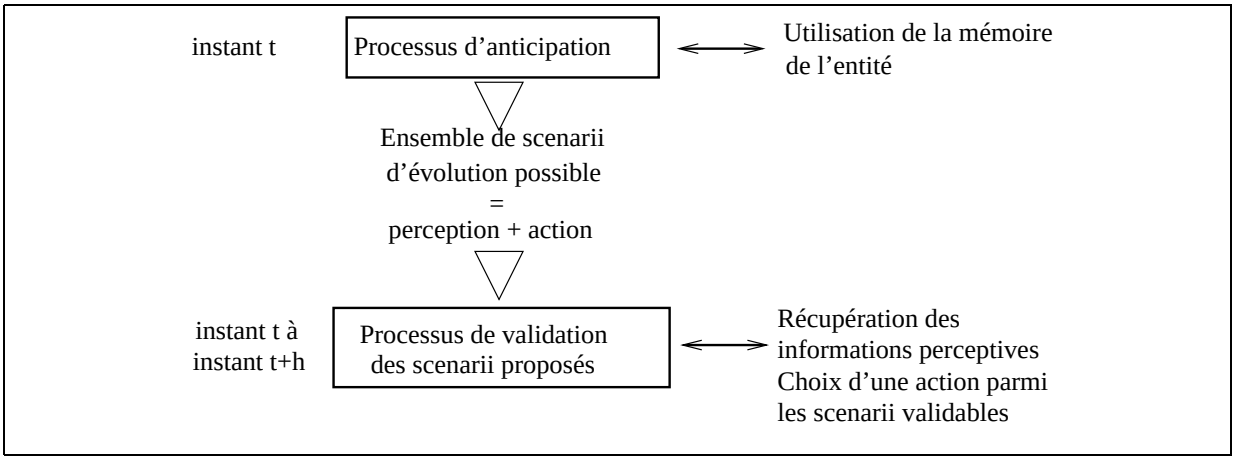


FIG. 5.2 – Schéma mettant en œuvre un processus d'anticipation

5.4 Importance des notions de fiabilité et de prédictibilité

5.4.1 Introduction

Au cours des deux sections précédentes, nous avons expliqué l'essence du processus de catégorisation et de l'AP. Il nous reste à établir les raisons de notre intérêt pour la fiabilité et la prédictibilité, ainsi que notre démarche visant à considérer l'apprentissage comme un résultat émergent de contraintes internes. Nous effectuons le lien entre les termes suivants :

- fiabilité et sensation de sécurité
- contraintes internes et cohérence entre la représentation mentale et le fait réel

Nous souhaitons également montrer que le réflexe sécuritaire exige la fiabilité, ce qui peut se faire au détriment de la recherche de l'optimalité.

5.4.2 Nature de la représentation mentale : cohérence entre l'anticipation et le fait observé

Il est clair que la plupart du temps d'un être humain est consacré à des activités routinières ; ce sont des tâches manuelles ou intellectuelles maintes fois répétées, qui sont devenues des automatismes au cours du temps. Ainsi, marcher, parler sa langue maternelle, faire du vélo, se déplacer vers un lieu connu, lire ou écrire sont des actes anodins à la réalisation desquels on ne prête pas d'attention au cours de leur exécution. Pourtant, ces tâches n'ont pas toujours été évidentes à accomplir : il y a eu apprentissage. Ce qu'il est intéressant de constater, c'est que chacune d'entre-elles est apprise parfaitement [1] dans tous les cas, du moins si la personne ne présente pas de troubles physiques spécifiques.

Nous prenons l'hypothèse selon laquelle l'apprentissage va de pair avec la construction d'une représentation mentale de la scène à apprendre. Le mot « scène » est ici employé d'une manière générale : il peut s'agir d'un comportement ou d'une catégorie perceptive à reconnaître.

1. Cela signifie que l'objectif associé à la tâche est atteint d'une manière certaine (en référence aux deux signaux émotionnels dont nous avons parlé à la fin du paragraphe 5.3.4). Toutefois, il ne s'agit pas de voir ici un quelconque caractère d'optimalité.

Il apparaît que l'apprentissage contient une phase durant laquelle les sens, mais aussi les organes moteurs, sont mis en éveil de manière à construire une réplique interne de la scène à apprendre. Celle-ci n'a pas pour but de construire un modèle objectif de la scène mais d'en extraire les traits caractéristiques, par rapport au but sous-jacent à l'apprentissage. Cela est réalisé bien sûr grâce aux sens à disposition de l'apprenant, mais aussi grâce aux moyens que possède celui-ci pour agir sur sa perception.

La représentation mentale peut être considérée comme l'image de l'interaction entre l'individu et son environnement vue sous l'angle d'un objectif particulier. Notre hypothèse est que cette image se construit en fonction de cet objectif, dans l'intention d'obtenir une cohérence entre cette image et l'expérience qui est effectivement vécue. La cohérence traduit la capacité à identifier la représentation mentale dynamique d'une scène (monde intérieur) avec la scène elle-même, vue au travers d'un objectif précis. Ainsi, une incohérence est mise en évidence dans le cas où le mécanisme interne de perception/action n'atteint pas son objectif. De ce point de vue, la perception de l'individu est dirigée à la fois vers l'extérieur (sur la scène réelle) et vers l'intérieur (pour faire « vivre » la représentation mentale de la scène, qui est orientée par l'objectif). Ces deux aspects doivent coïncider, lorsque l'apprentissage est réussi. Cela signifie qu'on arrive à créer cette représentation mentale et qu'elle suffit à orienter l'exécution d'un ensemble d'actions menant à l'objectif.

Comme nous le faisions remarquer au début de cette sous-section, les actes routiniers sont les plus fréquents chez une personne adulte ; d'après notre définition de l'apprentissage, la routine se produit lorsqu'il y a une cohérence parfaite entre les visions intérieure et extérieure d'une scène, donc sans apprentissage. Dans ce cas, la manipulation de la représentation mentale, qui est à la fois perception et action, permet à elle seule d'effectuer la tâche routinière ; la perception extérieure de la scène ne vient que confirmer l'exactitude de la dynamique de l'objet mental. Par contre, lorsque la scène n'est pas maîtrisée parfaitement, il se peut qu'il se produise une incohérence entre les visions intérieure et extérieure. Cela signifie que la représentation mentale de la scène n'est pas satisfaisante puisqu'un événement vient la contredire ; dans ce cas, l'apprentissage permet de modifier la représentation mentale de la scène afin que celle-ci soit de nouveau en cohérence avec le fait qui vient de se produire.

Contrairement à l'attitude objectiviste qui consiste à analyser l'environnement séparément des moyens de perception en en dégageant des régularités propres (création de modèles abstraits) puis à intégrer après coup les moyens de perception, la recherche de cohérence suppose une étude globale de la relation entre l'environnement et les moyens de perception.

5.4.3 Sensation de sécurité comme moteur de l'apprentissage

Comment la notion de cohérence se traduit-elle? Dans ce travail, nous supposons que le moteur de l'apprentissage est la recherche de la sécurité ou du bien-être. Ce point de départ est relativement comparable à celui des travaux de Pavlov, qui sont eux-mêmes l'origine « biologique » des méthodes d'apprentissage par renforcement. Mais, nous ajoutons la notion de certitude : cela signifie que l'objectif de l'apprentissage est l'atteinte à coup sûr d'un objectif.

Cette nouvelle exigence est extraordinairement forte et impose directement des restrictions sur la nature et la précision de l'objectif à atteindre. La figure 5.3 montre la différence entre l'approche que nous avons adoptée (schéma de droite) et l'approche classique, orientée uniquement

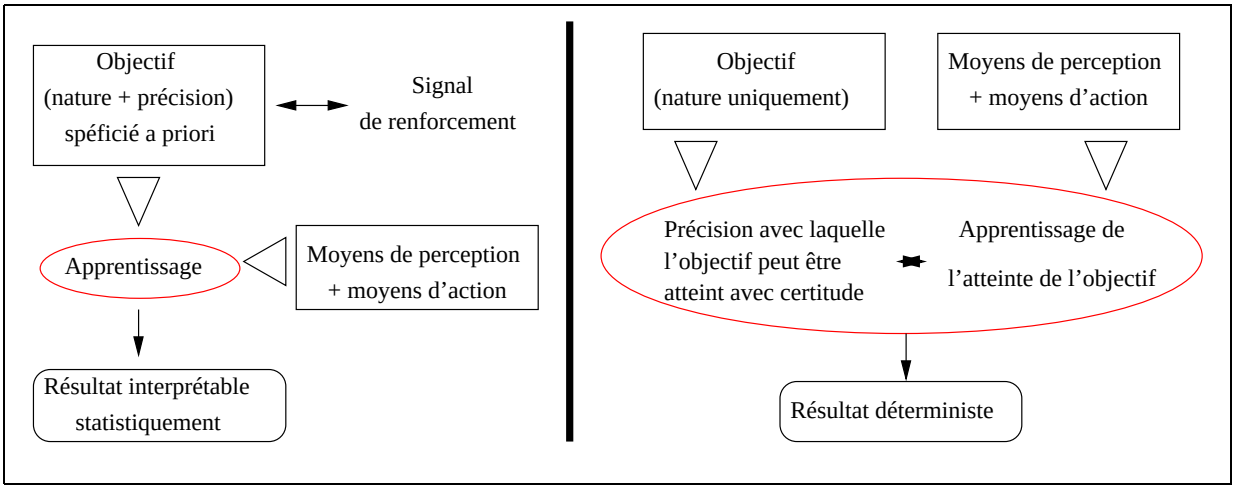


Fig. 5.3 – Démarches comparées d'atteinte d'objectif

par un objectif. Ainsi, nous distinguons la nature de l'objectif (ce qui doit être fait) avec le degré de précision avec lequel il doit être atteint. En effet, cette précision dépend essentiellement des moyens que l'individu possède pour résoudre le problème. En fait, cette distinction importante repose sur la question suivante : les êtres vivants recherchent-ils la meilleure solution à leurs problèmes (optimisation) ou se contentent-ils de solutions satisfaisantes (respectant un certain nombre de contraintes perceptibles) ? Nous posons ainsi le problème de l'adaptation de l'objectif lui-même aux possibilités de l'individu. En ce sens, nous sommes proches des idées développées par [Thagard et Barnes, 1996] et [Thagard et Millgram, 1997] dans l'atteinte d'objectifs de haut niveau [1] : **l'idée est qu'il est impossible de figer tout à fait un objectif**, car il risque de s'avérer incompatible avec la situation ou les possibilités de l'individu, conduisant à un échec ; il est donc nécessaire d'entrevoir un système d'aller/retour entre la précision (l'exigence) de l'objectif et les possibilités réelles de la personne. C'est pourquoi nous envisageons l'atteinte d'objectif comme le respect d'un ensemble de contraintes.

Notre choix implique que la recherche de la meilleure solution possible (optimalité) n'a pas de sens, en général, si elle est déconnectée des capacités qui existent effectivement pour obtenir cette solution. Or, les algorithmes d'apprentissage, fondés sur des méthodes d'optimisation, sont basés implicitement sur l'indépendance de la spécification de l'objectif et des moyens pour y parvenir. De plus, les techniques d'optimisation imposent certaines exigences. En particulier, il doit exister un référent unique (une fonction de coût), indépendant des moyens de perception et d'action de l'entité, qui permet finalement de comparer les solutions entre elles grâce à une distance. Or, ce principe de comparaison semble biologiquement peu plausible. Les patients étudiés par Damasio (voir la section 5.3.4) en sont un exemple. Privés du filtrage émotionnel, ils en sont réduits à évaluer chacune des solutions possibles en termes de coût, pour arriver à prendre la meilleure décision : cette technique aboutit à un échec. D'autre part, la notion de distance implique qu'il y ait transitivité : si x est préféré à y et y est préféré à z alors x est préféré à z. Cette propriété, supposée vraie dans la théorie de la décision classique, est souvent violée par les êtres humains ; c'est ce que montrent les études menées dans [Kahneman et Tversky, 1979] et [Tversky et Kahneman, 1981]. Dans le même ordre d'idées, la faculté d'attribuer une « note » à

1. Il s'agit de problèmes où l'objectif apparent peut induire des situations de dilemme, n'aboutissant *a priori* à aucune solution. Mais la transformation de l'objectif initial en un objectif moins ambitieux peut débloquer la situation en engendrant des possibilités de réalisation.

un fait, donc de pouvoir après coup ordonner ces faits, est très subjective, même si la notation est régie par des prédicats objectifs. Ainsi, une étude menée à Lyon sur la notation de devoirs de mathématiques donnés à des élèves de niveau DEUG a montré que, pour une même copie, des professeurs pouvaient donner des notes dont l'écart est de plusieurs points, alors qu'un barème précis au demi-point leur avait été fixé. D'autre part, un professeur pouvait noter la même copie a des moments différents avec des écarts conséquents.

Tout cela ne signifie pas que l'idée d'optimisation doit être exclue ; celle-ci tient une place importante, mais à un beaucoup plus haut niveau de l'intelligence : en effet, elle se traduit par la volonté consciente de l'individu d'élaborer une stratégie ou un algorithme précis pour obtenir un résultat le plus performant possible. Or, comme nous l'avons évoqué, l'apprentissage se déroule principalement à un bas niveau. L'hypothèse des marqueurs somatiques formulée par Damasio suggère qu'un filtre émotionnel permet aux possibilités pertinentes ou « bonnes » de parvenir au conscient de l'individu lors d'une prise de décision, en écartant les « mauvaises ». Les solutions jugées « bonnes » peuvent, après coup, être traitées consciemment avec une volonté d'optimisation. On voit apparaître l'opposition manichéenne entre le bon et le mauvais. Dans ce cadre, une solution doit pouvoir être jugée d'une manière binaire ; toutes celles qui sont étiquetées à « bon » sont donc des solutions équivalentes : elles respectent les contraintes du problème. En suivant cette hypothèse, la notion de distance entre deux solutions est donc rendue triviale.

Nous avons vu au début de ce paragraphe que l'opposition entre optimisation et satisfaction de contraintes met en lumière une deuxième question : « Comment gérer l'incertain ? ». Le formalisme de l'AR, utilisant les chaînes de Markov, donne une réponse à cette interrogation : le choix d'une action est fait par rapport à la fréquence des bonnes réactions qui ont été enregistrées dans le passé. Ainsi, l'individu cherche à maximiser ses chances d'effectuer la meilleure séquence d'actions possible (suivant un objectif précis) en utilisant son expérience passée, sachant qu'il se peut qu'il se trompe complètement dans son choix à un moment donné. Au contraire, l'objectif de respect de contraintes est de choisir une séquence d'actions dont l'individu est certain qu'elle ne mènera pas à un non respect de ces contraintes.

D'un point de vue comportemental, une solution visant l'optimal pourrait être qualifiée de « raisonnée », alors que le problème de respect de contraintes est « sécuritaire ». Comme nous l'avons mentionné au début de cette section, le premier cas est lié davantage à un comportement de « haut niveau », alors que le second est plus instinctif. En outre, le second point de vue est connecté à la notion biologique d'homéostasie, c'est-à-dire la caractéristique propre à tous les êtres vivants d'auto-réguler inconsciemment leur métabolisme. En effet, ce dernier peut être vu comme un ensemble de contraintes physiologiques internes à réguler impérativement (fréquence cardiaque, fréquence respiratoire, température, taux de glycémie, etc.). En outre, il existe une interaction entre le circuit homéostatique et les émotions [Damasio, 1999].

5.4.4 L'optimisation vue comme une capacité à générer un grand nombre de catégories perceptives

On pourrait objecter que la notion sécuritaire concerne les fonctions vitales de l'individu et que l'optimisation pourrait s'appliquer à d'autres fonctions : l'exemple du sportif de haut niveau semble aller dans ce sens.

Nous répondrions que nous sommes en accord avec cette objection, si l'optimisation est considérée comme le résultat de l'expérience humaine et non comme le moteur de l'apprentissage. Ce que nous supposons est que l'apprentissage n'est pas guidé en lui-même par des lois supposant l'optimisation d'un certain critère : il nous semble que l'objection présentée n'est pas de nature à contredire la vision de l'apprentissage comme le maintien de la cohérence entre la représentation mentale de l'individu et les faits réels. La notion d'optimisation employée dans notre exemple signifie, à notre avis, que la personne possède des capacités perceptives accrues, qui se traduisent dans notre système d'AP par **la possibilité de générer un grand nombre de catégories perceptives**, tout en vérifiant les contraintes d'observabilité et d'unicité.

Nous avons montré dans la deuxième partie du document que ce nombre est toujours fini, et qu'il dépend à la fois de la qualité des organes de perception et de l'ensemble d'hypothèses contenues dans la mémoire. Cependant, le fait qu'il existe un nombre maximum résulte du respect des contraintes CO et CU. Le terme « optimisation » est donc mal choisi dans ce cas : nous préférons le remplacer par celui de **précision**.

Références

Abott, E. (1952). *Flatland. A Romance in Many Dimensions.* Dover Publications, New York.

Barlow, H. (1985). The role of single neurons in the psychology of perception. *Q.J. Expl Psychology*, 37A :121–145.

Barlow, H. (2001a). The exploitation of regularities in the environment by the brain. *Behavioral and Brain Sciences*, 24(3).

Barlow, H. (2001b). Redundancy revisited. *Network : Computation in Neural Systems*, 12 :241–253.

Barron, J., Fleet, D., and Beauchemin, S. (1994). Performance of optical flow techniques. *IJCV*, 12 :43–77.

Cannon, W. (1932). *The Wisdom of the Body.* Norton, New York.

Damasio, A. (1994). *Descartes'Error : Emotion, Reason and the Human Brain.* Picador.

Damasio, A. (1999). *Le sentiment même de soi - Corps, émotions, conscience.* Editions Odile Jacob Sciences.

Eimas, P., Siqueland, E., Jusczyk, P., and Vigorito, J. (1971). Speech recognition in infants. *Science*, 171 :303–306.

Fantz, R., Ordy, J., and Udelf, M. (1962). Maturation of pattern vision in infants during the first six months. *Journal of Comparative Physiological Psychology*, 55 :907–917.

Farah, M. (1992). Is an object an object ? cognitive and neuropsychological investigations of domain-specificity in visual object recognition. *Curr. Dir. Psychol. Sci.*, 1.

Gibson, E. (1969). *Principles of Perceptual Learning and Development.* Appleton, Century and Croft.

Gibson, J. and Gibson, E. (1955). Perceptual learning : Differentiation or enrichment ? *Psychological Review*, 62 :32–41.

Goldstone, R. (1998). Perceptual learning. *Annu. Rev. Psychol.*, 49 :585–612.

Hanneton, S., MacIntyre, J., and J., D. (1998). Contribution of haptic and predictive cues to adaptive manual control of a dynamic system. *Motor Control*.

Kahneman, D. and Tversky, A. (1979). Prospect theory : An analysis of decision under risk. *Econometrica*, 47 :263–291.

O'Regan, J. and Noé, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5).

O'Regan, J., Rensink, R., and Clark, J. (1999). Blindness to scene changes caused by mud-splashes. *Nature*, 398.

Sauvage, G. (1999). *Les marchés financiers. Entre hasard et raison : le facteur humain*. Seuil.

Teller, D. (1981). The developement of visual acuity in human and monkey infants. *Trends of Neurosciences*, pages 21–23.

Thagard, P. and Barnes, A. (1996). Emotional decisions. *Proceedings og the Eighteenth Annual Conference of The Cognitive Science Society*, pages 426–429.

Thagard, P. and Millgram, E. (1997). Inference to the best plan : A coherence theory of decision. *Goal-Driven Learning*, pages 439–454.

Tversky, A. and Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science*, 211 :453–458.

Werker, J. and Lalonde, C. (1988). Cross-language speech perception : Initial capabilities and developmental change. *Developmental Psychology*, 24(5) :672–683.

6

Positionnement de notre démarche scientifique

6.1 Introduction

Dans le chapitre précédent, nous avons esquissé une explication informelle des fondements de notre démarche scientifique en puisant des exemples dans le domaine du vivant. Cela apporte une certaine assise à notre discours, en montrant pourquoi il a été construit. Cependant, le travail de justification de notre démarche ne serait pas complet s'il n'esquissait pas une réponse aux deux questions suivantes :

- en quoi notre démarche se différencie-t-elle des démarches scientifiques existantes ?
- notre postulat suppose l'existence de lois appliquées au vivant, donc qui ont une action dans le monde réel. Quelle notion de la réalité cela suppose-t-il ? Y-a-t-il une correspondance avec une ou des notions de la réalité déjà exprimée dans des contextes scientifiques différents du nôtre ?

Pour tenter de répondre à la première question, il nous semble intéressant de revenir sur quelques démarches essayant de traiter le problème de l'apprentissage. D'un point de vue historique, Turing pose le problème général de la possibilité d'imiter l'intelligence par des machines [Turing, 1950]. Le *test de Turing* se veut être un critère objectif des capacités d'imitation. Nous pensons que le dispositif expérimental de Turing est biaisé, parce que l'observateur est, dans son cas, juge et partie : celui-ci interprète les résultats donnés soit par l'humain, soit par la machine, en utilisant ses propres connaissances. **Ce raisonnement est à comparer avec celui que nous avons effectué dans le cadre du problème "D", pour lequel l'observateur n'utilise aucune connaissance particulière**.

Le deuxième point pose une question très ardue, à laquelle un début de réponse peut être esquissé à la lumière de la vision de la réalité développée par le physicien d'Espagnat ([d'Espagnat, 1985], [d'Espagnat, 1994]). Nous souhaitons montrer quelques points de similitude entre la notion de réalité, que notre démarche implique, et celle développée par d'Espagnat. Encore une fois, il ne s'agit nullement de prouver la validité de notre démarche, mais simplement d'appuyer notre travail en lui donnant davantage de crédibilité.

Ce chapitre cherche donc à montrer le bien-fondé du principe même de notre travail, en développant une réflexion très générale (et forcément très incomplète) sur les courants de pensée scientifiques utilisés pour travailler autour de la notion d'intelligence et de la notion de réalité. La

première section traite de l'approche de l'intelligence. Nous la concluons en précisant comment notre approche peut s'inscrire dans ce débat. La deuxième section ébauche une réflexion sur le concept de réalité, que notre approche sous-tend.

6.2 Différentes approches de l'Intelligence

6.2.1 Introduction

L'objectif de cette section est d'effectuer un rapide tour d'horizon des concepts précurseurs des différentes approches de l'intelligence. Bien entendu, il ne s'agit pas de détailler chacune des méthodologies, ni l'ensemble des résultats auxquels on est parvenu en les employant. Nous n'avons pas non plus l'intention d'aborder formellement la technicité propre à chacun de ces problèmes. En particulier, nous n'avons pas le souci d'être exhaustifs sur le « Comment faire? » associé à chaque voie de recherche. Mais, nous souhaitons principalement mettre en lumière le contexte associé à chacune de ces approches, qui a servi de base à la construction de notre étude. En particulier, il est intéressant de constater que les approches fonctionnalistes de l'intelligence, qui sont en grande partie déconnectées de la réalité biologique du cerveau, permettent de créer des programmes imitant de mieux en mieux les aspects du comportement humain qu'on pourrait qualifier d'intelligents, en étant, en contrepartie, prisonniers d'un contexte d'utilisation très précis (peu adaptatifs). D'un autre coté, les approches plus centrées sur la faculté d'adaptation et biologiquement ou cognitivement plus plausibles ne prétendent aucunement vouloir reproduire mécaniquement les facultés humaines ou sont limitées quant à l'envergure du problème qu'elles sont capables de traiter.

6.2.2 Idées fondatrices associées à la machine de Turing - hypothèse béhavioriste

Les ordinateurs actuels sont énormément plus rapides que le premier calculateur *Colossus*, créé en 1943 avec l'aide de Turing dans le but de déchiffrer rapidement les messages codés allemands. Cependant, ils découlent tous du concept de la machine de Turing [Turing, 1936]. Celle-ci a été proposée par Turing en réponse à une question ouverte posée par le mathématicien allemand Hilbert [Hilbert, 1928] concernant la possibilité de démontrer mécaniquement des théorèmes mathématiques.

Selon la conjecture de Turing, une machine de Turing peut effectuer n'importe quelle tâche pouvant être exécutée de manière purement mécanique. Une tâche ou procédure ou méthode T est dite "mécanique" si elle vérifie les quatre assertions suivantes :

- T est décomposable en un nombre fini d'instructions exactes (dont l'action est précisément connue), chaque instruction pouvant être exprimée grâce à un nombre fini de symboles ;
- Si aucune erreur technique ne survient, T produit invariablement le résultat désiré en un temps fini ;
- T peut être exécutée (en pratique ou en principe) par un homme, sans avoir recours à aucun accessoire exceptés un crayon et un papier ;
- T ne requiert aucune intelligence ou connaissance spécifique à l'homme qui l'exécute.

En outre, grâce à une règle d'évolution interne, la machine de Turing est capable de déterminer parfaitement son évolution, étape par étape, connaissant son état présent et l'instruction

L'automate de Turing possède un nombre d'états fini et est composé d'une tête devant laquelle se déroule un ruban divisé en cellules et qui peut se déplacer aussi bien de droite à gauche que de gauche à droite. N'importe quelle fonction calculable peut être calculée par cet automate grâce à l'ensemble de règles auxquelles il est soumis et que l'on peut toujours déterminer. Ces règles sont de trois types:

1) Si Aj Bm Sk
Dans un état Si, l'automate remplace Aj, contenu dans la cellule au dessous de la tête par Bm, puis passe dans un état Sk

2) Si Aj DSk
Le ruban est déplacé d'une cellule vers la droite et l'automate passe à l'état Sk

3) Si Aj GSk
Le ruban est déplacé d'une cellule vers la gauche et l'automate passe à l'état Sk

FIG. 6.1 – Principe de la machine de Turing

exécutée (voir la figure 6.1).

Le problème qui n'est pas abordé par la conjecture de Turing est de savoir si le fonctionnement d'une entité intelligente particulière peut être décrit en terme de "tâche mécanique". Cette question apparaît en fait comme le point de départ de la recherche en intelligence artificielle, si on considère l'intelligence comme un principe dont on essaie d'expliquer le fonctionnement : l'intelligence peut-elle être réduite à une succession d'instructions élémentaires ?

Dans [Turing, 1950], Turing élabore un protocole expérimental, connu sous le nom de "test de Turing", dans lequel la faculté d'intelligence d'une machine est évaluée selon des critères comportementaux : pour un problème donné (dans ce cas précis, il s'agit de tenir une conversation en anglais), si un juge humain n'est pas capable de différencier la réponse de la machine par rapport à la réponse d'un autre homme, la machine est déclarée "intelligente" (pour ce problème précis). Par là-même, Turing fait l'hypothèse béhavioriste selon laquelle l'intelligence peut se juger uniquement en termes de résultats observables, c'est-à-dire qu'elle se réduit à la capacité de produire certains comportements qu'un observateur humain qualifierait d'intelligents ; en particulier, cela exclut toute forme d'intentionnalité (inobservable de l'extérieur, par nature). Cette hypothèse a été critiquée à maintes reprises (voir par exemple le point de vue de Edelman dans [Edelman, 1992]). En particulier, il semble que, pour Turing, l'intelligence est uniquement la capacité à raisonner et à formuler ce raisonnement. Dans ce cas précis, le test de Turing a été réussi (au moins en partie) (voir par exemple le système ELIZA [1] [Weizenbaum, 1976]). Des réussites remarquables ont été obtenues dans le domaine de l'expertise médicale (Mycin [Shortliffe et Buchanan, 1975]) ainsi que

1. ELIZA est un programme simple qui imite un psychanalyste face à son patient. ELIZA est un exemple de programme qui réussit le test de Turing, sans pour autant implémenter des techniques d'Intelligence Artificielle

dans les jeux (programmes capables de battre les meilleurs joueurs mondiaux [Hsu et al., 1990]). Donc, il apparaît que, selon l'acception de l'intelligence choisie par Turing, des machines peuvent faire preuve d'intelligence, dans un domaine précis. Et, en suivant la conjecture que l'intelligence est une somme finie de comportements jugés intelligents, on pourrait construire les éléments d'une entité intelligente. C'est là un raisonnement classique de résolution de problèmes : découper un tâche compliquée en un ensemble fini d'opérations plus simples, résoudre celles-ci séparément puis assembler les résultats. Nous verrons par la suite que cette démarche est inopérante dans certains cas.

Sans doute, la volonté de Turing est d'appliquer une vraie rigueur scientifique à l'étude de l'intelligence : celle-ci est considérée comme un phénomène observable. L'approche de l'intelligence que Turing adopte est dualiste, dans la lignée de la pensée cartésienne, dans le sens où le phénomène « intelligence » est détaché de son contenant (l'être vivant) [1]. Ainsi, le test de Turing impose la présence d'un observateur, qui doit être neutre par rapport à ce qu'il étudie dans le sens où il n'interfère pas sur les objets de l'expérience, ni sur le résultat de celle-ci. Cela signifie en particulier que l'intelligence peut être définie à partir d'un référentiel unique, indépendant du "contenant", c'est-à-dire de l'entité qui manifeste de l'intelligence. En particulier, celle-ci est dissociée de toute subjectivité.

Cependant, deux grandes critiques peuvent contredire sensiblement la vision de Turing.

- D'une part, l'intelligence ou la raison, telle que la conçoit Turing, dans le cas où elle serait réductible à un programme implanté dans une machine de Turing, serait une entité intemporelle, c'est-à-dire un résultat fini, sans évolution. Cependant, lorsque Turing pose le problème "Les machines sont-elles capables de penser ?" [Turing, 1950], il admet que la programmation de tous les éléments d'une machine pensante est une tâche trop gigantesque et qu'il serait souhaitable que certaines composantes soient apprises ; de ce fait, il suppose que cette machine puisse évoluer, par apprentissage, pour acquérir certains comportements, lui conférant des propriétés dynamiques d'évolution.
- D'autre part, l'intelligence est considérée comme une entité auto-suffisante, désincarnée en fait : l'influence des organes sensoriels ou moteurs est négligée. Mais comment pourrait-il en être autrement dans un contexte où la subjectivité est écartée *a priori* ? En effet, la nature biologique de la perception peut certes être décrite objectivement, mais ce n'est pas le cas si la perception est vue comme un outil ou une fonction utilisée par l'intelligence. En d'autres termes, si on se réfère à la définition de l'intelligence en tant que tâche mécanique, comment associer d'une manière exacte une perception particulière à un symbole définissant une partie d'une instruction ? Turing comprend que cette question est importante et propose une voie de recherche allant dans ce sens, en conclusion de son article ; d'après lui, une possibilité d'évolution de l'intelligence artificielle serait d'équiper un ordinateur des meilleurs organes sensoriels possibles et de leur apprendre à parler et à comprendre l'anglais (dans le cadre du test de Turing), c'est-à-dire apprendre le passage de la perception à la compréhension de celle-ci, donc à sa forme symbolique. Cependant, la mécanique d'apprentissage n'est pas abordée par Turing.

L'introduction de l'apprentissage dans le développement de Turing est donc une réponse à

1. Néanmoins, comme l'a fait Descartes, Turing pose le problème de « l'interfaçage » entre la pensée abstraite et le corps en pratique, car il convient de la nécessité de pourvoir une machine d'éléments de perception et d'action sur son environnement, afin de reproduire les capacités humaines. Mais il s'avère que l'interfaçage n'est pas un problème mineur, à l'inverse de ce que pensait Turing

deux problèmes pratiques majeurs :

- la multiplicité des cas : la manifestation du phénomène "intelligence" est trop compliquée pour en prévoir tous les cas. Lorsqu'un nouveau cas apparaît, il faut donc l'apprendre.
- la gestion de l'interface entre la machine et son monde externe : l'interaction entre la machine et ses organes de perception doit être apprise.

Donc, de l'aveu même de Turing, la faculté d'apprentissage est une composante nécessaire dans l'établissement du programme "intelligence" destiné à être exécuté par une machine de Turing. Cependant, il est clair que le problème initial supposant l'existence d'un programme traduisant le phénomène d'intelligence n'est pas résolu par cette constatation. Bien au contraire, il nécessite l'apport d'une autre conjecture portant sur la nature de l'apprentissage. Pourtant, le problème est d'une autre nature que celui portant sur l'intelligence : cette dernière utilise ou transforme des faits (ou des données) pour en produire de nouveaux, que ceux-ci soient objectifs ou subjectifs ; par contre, l'apprentissage d'un comportement est le résultat d'une modification interne de celui-ci. Une "modification interne" peut être engendrée *a priori* par deux catégories de changements :

- l'évolution des valeurs des paramètres (données) utilisés par les instructions codant le comportement
- la modification des instructions codant le comportement

Dans le premier cas, le comportement à apprendre peut être considéré comme une fonction paramétrique dont les paramètres évoluent au cours de l'apprentissage, sans pour autant modifier la séquence d'instructions programmant ce comportement ; les algorithmes d'apprentissage des réseaux neuronaux artificiels suivent cette approche, dans un cadre toutefois très différent de celui de la machine de Turing (voir la section suivante).

Par contre, dans le second cas, l'apprentissage signifie une modification du programme lui-même. Cette démarche est beaucoup plus complexe que la première. Deux approches conceptuellement très différentes vont dans ce sens :

- Dans le cadre de l'Intelligence Artificielle, l'auto-enrichissement de systèmes à base de connaissances[1] peut être considérée comme un apprentissage par modification d'une partie du programme. Des systèmes sont capables de découvrir de nouvelles connaissances ou de nouvelles règles à partir, par exemple, de méta-règles ou de méta-connaissances [Pitrat, 1990]. Il s'agit bien d'un cas d'apprentissage puisque, suite à la découverte d'une nouvelle règle ou d'une nouvelle connaissance, le système sera capable le cas échéant de générer de nouveaux faits, donc de modifier une partie de son comportement.
- Dans le cadre de l'application des algorithmes génétiques, un mécanisme de sélection choisit les programmes les plus performants (suivant un référentiel donné, la fonction de *fitness*), peut les croiser et former un nouveau programme. La notion de connaissance apprise n'apparaît pas directement ici, contrairement au cas précédent. Par conséquent, il n'y a pas non plus de notion d'enrichissement de la mémoire par de nouvelles connaissances.

1. Il faut différencier un fait d'une connaissance. Le premier est une donnée qu'on utilise dans un cadre précis, alors que la connaissance s'utilise "selon un mécanisme qui suit les indications données par la connaissance pour aboutir finalement à un résultat" ([Pitrat, 1990], page 30)

6.2.3 Idées associées à l'approche cognitive de l'intelligence

Alors que l'approche béhavioriste se concentre sur le résultat - observable de l'extérieur - de l'intelligence, admettant qu'il existe une séparation nette entre l'esprit (non exprimable scientifiquement) et sa manifestation (seul objet d'étude), l'approche cognitive tente d'effacer ce dualisme ; en particulier, le référent n'est plus le résultat mais le sens que l'on peut donner à ce qui est perçu ou à un comportement.

La possibilité de donner un sens à ce qui est perçu (la perception étant employée dans le sens le plus large) nécessite en outre la faculté de catégorisation.

Comment le sens naît-il à partir de la perception ? Une hypothèse assez répandue consiste à considérer une représentation sémantique de la signification. Pour cela, la représentation mentale est supposée être abstraite. En particulier, on admet l'existence d'un ensemble fini de symboles qui sont associés à des catégories classiques, choisies et fixées *a priori*. Par « catégories classiques », nous entendons celles dans lesquelles l'appartenance est définie d'après des conditions nécessaires et suffisantes. En suivant cette voie, si on parvient à construire pour chaque perception une association avec une catégorie, donc un symbole, celui-ci peut être employé au sein de règles logiques ou, plus généralement, de calculs. Somme toute, les approches béhavioristes rejoignent celle-ci dans la possibilité de construire un système logique pouvant représenter l'intelligence. Dans ce cadre, nous avons un système possédant deux niveaux hiérarchiquement connectés : le premier est chargé d'effectuer l'interface entre le monde extérieur (les perceptions de l'entité) et le monde intérieur, en associant la perception avec un symbole, alors que le second utilise ce symbole « désincarné » au sein d'un ensemble de calculs (voir la figure 6.2) . La démarche de catégorisation utilisant un ensemble figé de symboles abstraits suppose en fait que :

- les objets du monde extérieur à une entité sont catégorisables exactement ;
- le nombre de catégories est fini et on est capable de les dénombrer d'une manière exhaustive ;
- la représentation mentale d'un objet est identique d'une personne à l'autre ;
- les catégories sont indépendantes du moyen dont elles sont perçues ;
- la mémoire d'un phénomène se résume à un ensemble de symboles et à la relation entre ceux-ci.

D'une certaine manière, on retrouve la vision fonctionnaliste, supposant un monde statique et fermé où tout est censé être connu (car catégorisable) d'un manière objective. En particulier, la perception est analysée, à l'interface entre le monde extérieur et le monde intérieur, par la correspondance avec un symbole dont l'existence est supposée être indépendante de la réalité physique de l'entité le manipulant.

Les faits ne remettent pas en cause l'approche symbolique (c'est-à-dire l'existence d'une faculté de catégorisation d'éléments perceptifs), mais l'existence d'une manière unique, transposable d'un individu à l'autre, d'appréhender un symbole donné. *A priori*, chaque individu crée sa propre représentation mentale d'un fait ou d'une scène, à partir de ses facultés de perception et d'action sur le monde. Par conséquent, l'objet mental dépend intimement des moyens qui l'ont généré, donc de l'individu qui l'abrite ; ce n'est pas une « substance » échangeable d'une personne à l'autre. En d'autres termes, si l'on veut être cohérent avec la réalité biologique, le problème de catégorisation ne peut pas être résolu en spécifiant *a priori* un certain nombre de symboles et en effectuant *a posteriori* une correspondance de signaux perceptifs avec l'un de ceux-ci. Dans le cadre de l'apprentissage, cela implique que les symboles soient créés par l'expérience, ce qui n'empêche pas, si besoin est, de leur attribuer une interprétation *a posteriori*. La

démarche de classification engendrée par l'algorithme des cartes auto-organisatrices de Kohonen [Kohonen, 2001] reproduit ce principe.

En robotique mobile, un exemple classique de catégorisation est celui de la reconnaissance ou de la reconstruction d'environnements. S'il s'agit d'un environnement du type « labyrinthe », une idée fréquemment usitée consiste à décrire le lieu en termes d'assemblage de symboles simples (couloir, coin, intersection, etc.) utilisés au sein de règles logiques [Pradel et Barret, 1998]. L'expérimentateur choisit cet ensemble de symboles car ils lui semblent (conformément à sa propre représentation du monde) être représentatifs d'un tel environnement et avoir chacun des propriétés géométriques particulières ; nous sommes ici dans une tentative de description objective d'une scène. Or, si le robot est équipé de capteurs frustres (infrarouge ou ultrason), voit-il le monde comme nous pouvons le faire ? Certainement pas. Les catégories que nous avons formées ont-elles un sens pour ce robot ? Nous ne pouvons pas *a priori* répondre à cette question. En fait, deux problèmes principaux surviennent, qui sont la cause d'ambiguïtés :

- il peut être difficile de discriminer deux catégories parce que les symboles sont mal choisis par rapport à la perception du robot (ils peuvent être vus d'une manière très proche par celui-ci)
- il n'existe pas de frontière nette entre deux archétypes (le passage d'un couloir à une intersection peut être reconnu comme « couloir » ou « intersection » à un instant donné)

Alors que la première cause d'ambiguïté est rédhibitoire (le problème est mal posé) mais évitable, la seconde est naturelle, du moins si on fait l'hypothèse que le robot passe continûment d'une situation perceptive (le couloir) à une autre (l'intersection), mais elle est inévitable. Toutefois, tel que le problème est posé, il semble qu'aucune méthode (directe ou par apprentissage) ne peut parvenir à l'élimination de l'incertitude qu'on peut avoir sur la véracité de l'association perception/symbole. On peut néanmoins obtenir un degré de certitude sur l'association fournie (réseaux de neurones bayésiens). Cela permet de regarder d'un œil critique la réponse avancée et de rejeter celle-ci le cas échéant. Mais ce degré de confiance n'est jamais nul (« je ne sais pas ») ou maximum (« je suis certain »).

L'avantage d'un passage au symbolique est clair : les symboles (abstraits) peuvent être manipulés aisément au sein de propositions logiques : « *Si le robot détecte tel symbole ou groupement de symboles (avec un degré de certitude élevé) alors il doit accomplir telle tâche* ». La gestion de la perception est alors une fonction de bas niveau, exploitée par un programme gérant la stratégie du robot (processus de haut niveau). Mais la détection du symbole ne peut s'exprimer qu'en termes statistiques, puisque l'ambiguïté inhérente à la démarche employée rend toute certitude impossible sur la validité de la reconnaissance. Qu'en est-il alors de l'exécution du programme formé de règles utilisant des symboles perceptifs ? Son résultat n'est pas garanti : au mieux, il est probablement exact.

À travers cet exemple, nous constatons que le fait d'admettre d'office l'existence d'un ensemble de symboles conduit à une démarche qui est confrontée à la gestion de l'incertain (voir la figure 6.3). Mais peut-on imaginer que celle-ci soit réaliste d'un point de vue psychologique ? Il est clair que l'incertitude fait partie intégrante de la vie humaine, mais certainement pas au niveau de la perception « courante » du monde extérieur : la marche est un processus complexe multi-modal (vue, toucher au niveau des pieds, centre de l'équilibre au niveau de l'oreille interne), faisant intervenir plus d'une cinquantaine de muscles ; pourtant, lorsque le bébé a appris à marcher, il ne tombe plus. Le sentiment de certitude dans l'accomplissement correct de gestes

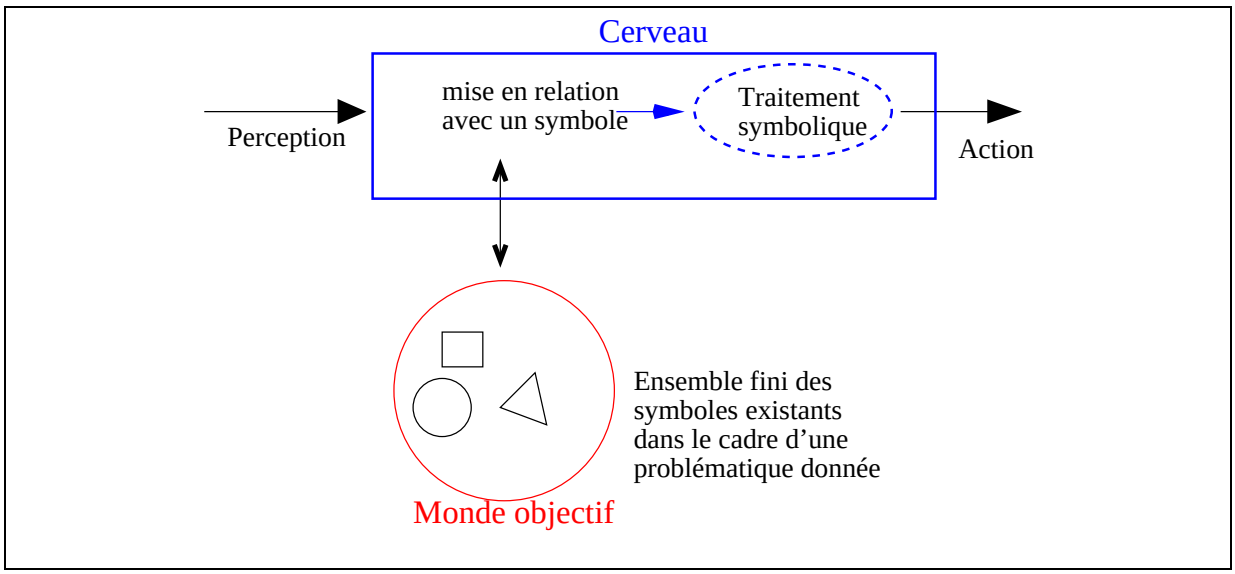


FIG. 6.2 – Une approche fonctionnaliste en sciences cognitives

réflexes aussi bien que dans les réponses du monde extérieur dans des situations familières est d'ailleurs un impératif pour l'être humain. Nous pensons que la gestion de l'incertain est un processus de haut niveau, conscient et raisonné, qui n'est pas concevable à l'échelle du traitement de la perception.

Mais la notion d'objet mental peut se concevoir sans l'utilisation d'un ensemble de symboles prédéfinis. Ainsi, Lecerf établit une théorie de l'objet mental - la double boucle de l'apprentissage cognitif - inspirée directement de considérations biologiques [Lecerf, 1997]. Selon son hypothèse, l'objet mental émerge d'un processus de résonance provoqué par une structure neuronale cyclique : l'influx sortant de chaque neurone du cycle peut en affecter un autre, engendrant une propagation auto-entretenue du signal ; l'objet mental serait alors caractérisé par la donnée des neurones touchés, mais également par le chemin parcouru par le signal nerveux, donc par une certaine dynamique existant à travers ces neurones. Lecerf fait un tour d'horizon des processus mentaux (mémorisation, apprentissage, abstraction, etc.) en les expliquant à la lumière de son schéma de l'objet mental.

Dans ce cas, l'ensemble des symboles représente toutes les configurations neuronales dans lesquelles un signal auto-entretenu peut être engendré ; ces configurations sont obtenues par apprentissage. Une image mentale particulière est donc activée par une perception donnée, qui est ensuite auto-entretenue dans le cerveau par un certain nombre de neurones (voir la figure 6.4).

Le problème est ici de trouver une loi d'organisation de l'architecture des neurones, dont le résultat est la création de cycles ayant des propriétés émergentes particulières (rendre compte des spécificités perceptives d'un symbole). Si une scène non connue est présentée, aucune configuration neuronale préétablie n'engendrera un signal auto-entretenu : cette propriété est intéressante, car elle permet la reconnaissance concrète de l'état d'incertitude. Celui-ci pourra être levé par apprentissage, donc par la constitution d'une nouvelle structure neuronale.

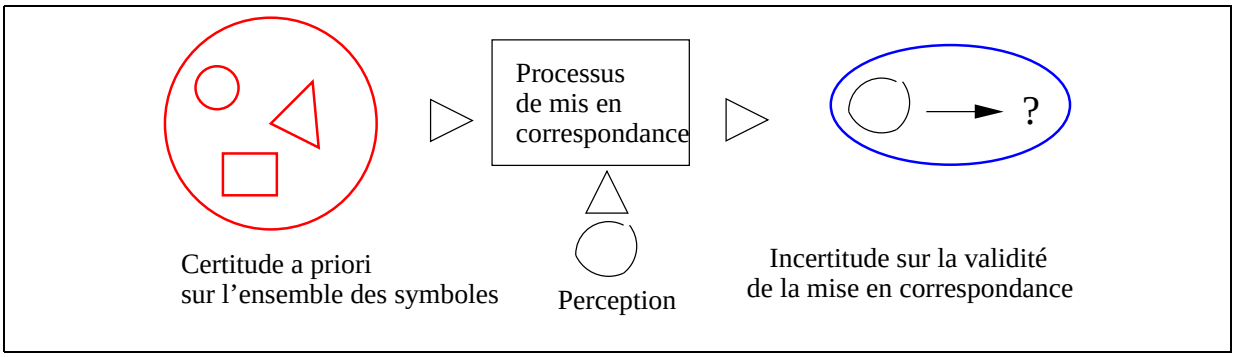
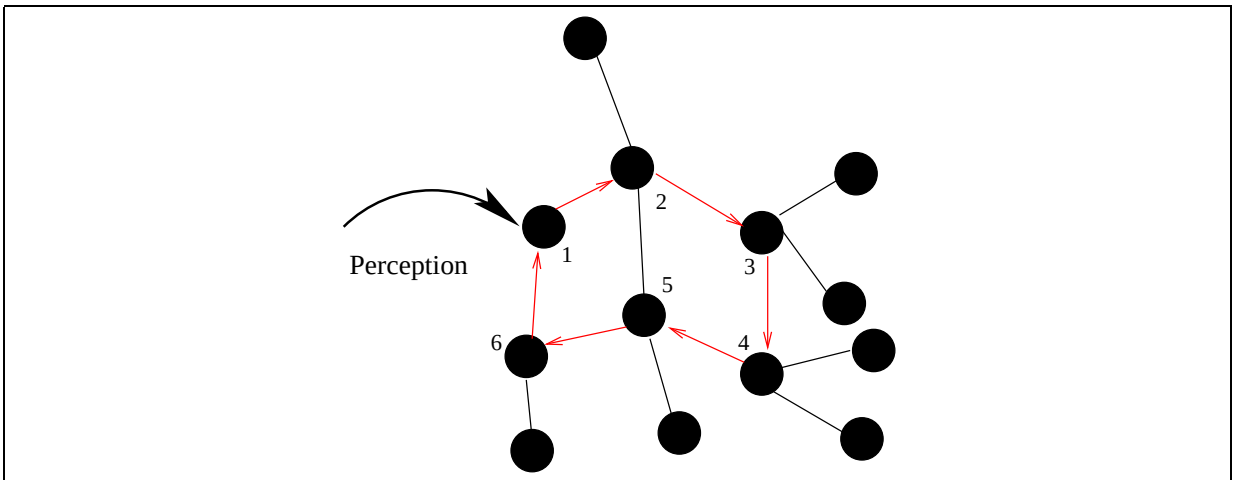


FIG. 6.3 – Une démarche confrontée à la gestion de l'incertitude



La perception est émise au niveau du neurone 1. Un signal électrique se propage à partir de ce neurone et donne naissance à un signal auto-entretenu sur les neurones 1 à 6, dans le cycle (1,2,3,4,5,6).

FIG. 6.4 – Le symbole vu comme auto-entretien d'un signal sur un cycle neuronal

6.2.4 Idées associées à l'approche biologique de l'intelligence

Une démarche parallèle à celle de Turing est l'étude du support de l'intelligence, c'est-à-dire le cerveau. En supposant qu'on comprenne le fonctionnement des éléments formant le cerveau, ainsi que l'interaction entre ces éléments, on pourrait saisir le comportement global de celui-ci, donc les caractères de l'intelligence. Très grossièrement, le cerveau peut être vu comme un énorme ensemble de neurones interconnectés dans lesquels passe un faible courant électrique. Historiquement, le premier réseau de neurones a été présenté par Bain [Bain, 1873]. Celui-ci propose une loi, reformulée par James [James, 1890] puis par le psychologue Hebb [Hebb, 1949], mettant en avant les facultés d'adaptation des connexions neuronales qui peuvent dans certains cas se renforcer et dans d'autres s'inhiber. Cette possibilité d'évolution, associée à l'idée d'apprentissage, est à la base des sciences connexionistes. L'autre idée forte est celle de l'émergence de phénomènes complexes résultant de l'interaction entre différents agents simples (les neurones). On retrouve cette idée dans le cadre de l'étude de systèmes coopératifs multi-agents.

Des approches de formalisation du neurone ont été faites dès la moitié du vingtième siècle. Ainsi, N. Rashevsky suggéra que le cerveau pourrait être vu comme une organisation complexe de 0 et de 1. Mais c'est en 1943 que Warren McCulloch et Walter Pitts [McCulloch et Pitts, 1943] mirent en évidence que la fonction d'activation d'un neurone pouvait être modélisée par une fonction à seuil, suivant les potentiels électriques arrivant à l'entrée du neurone. L'ensemble des neurones connectés forme un réseau de neurones artificiel. Le modèle du neurone artificiel, utilisant les propriétés biologiques des neurones et de leur interaction, a été amélioré à de nombreuses reprises. Rosenblatt en a été le précurseur en créant le Perceptron [Rosenblatt, 1958]. En fait, dans le cas où les connexions entre neurones sont figées, le réseau de neurones artificiel n'est rien d'autre qu'une fonction paramétrique dont les paramètres principaux sont les poids synaptiques. Par conséquent, un réseau de neurones multi-couches tel que le Perceptron n'est rien d'autre qu'une boîte noire possédant des entrées et des sorties dont le nombre est fixé *a priori*. La modification des paramètres internes par une règle d'apprentissage supervisée (méthode de rétropropagation du gradient [Rumelhart et al., 1986]) permet à cette fonction paramétrique d'interpoler avec un minimum d'erreur un ensemble d'exemples d'association entrée/sortie.

Mais le modèle du réseau de neurones artificiel, bien qu'il ait été conçu à la base grâce à des observations biologiques, est très éloigné du réseau de neurones biologique. En particulier, le fait que les connexions entre neurones et que l'organisation des couches soient figées le rend particulièrement irréaliste d'un point de vue biologique. De plus, dans ce cas, le terme « apprentissage » signifie simplement « interpolation avec un minimum d'erreur ».

La connaissance du fonctionnement d'un neurone isolé n'est pas suffisante pour comprendre l'émergence de phénomènes intelligents. La manière dont ceux-ci sont connectés et se regroupent en zones spécialisées dans le traitement de certaines tâches permet de donner une « carte géographique » du cerveau. En particulier, les zones impliquées dans l'exécution de tâches courantes (régulation homéostatique, reconnaissance d'un visage, écoute, marche, etc.) sont connues assez précisément. A partir de ces faits, l'étude de l'organisation et des mécanismes de certaines zones du cerveau a inspiré la constitution d'algorithmes d'apprentissage élaborés (voir les travaux effectués entre autres au laboratoire ETIS [Revel, 1997]). D'une manière générale, l'objectif est de reproduire le plus fidèlement possible une structure neuronale complexe particulière, connaissant sa fonction biologique, de manière à générer un fonctionnement artificiel accomplissant une tâche proche de celle générée par la structure réelle. Cette démarche est identique à celle des pionniers

de l'aviation tentant de construire une machine volante imitant l'oiseau.

Mais la complexité extrême du cerveau rend naturellement la tâche quasi-impossible dans sa totalité. On est alors amené à se concentrer sur un mécanisme particulier, une zone restreinte, en perdant la vision globale du fonctionnement du cerveau ; en fait, on est poussé à adopter une stratégie similaire à celle décrite dans le paragraphe 6.2.2, pour des raisons identiques à celles de Turing lorsqu'il avouait la nécessité d'un système d'apprentissage. De fait, si on veut étudier un système biologique dans sa globalité pour en traduire le fonctionnement d'une manière mécanique, on est obligé d'abandonner l'humain et de se centrer sur l'étude d'animaux supposés beaucoup plus simples, comme la fourmi. Mais, la complexité du problème est-elle réduite pour autant ? En ce qui concerne l'étude des fonctionnalités intrinsèques à une fourmi, c'est certain. Des « animats » apprennent à marcher et à se déplacer comme une fourmi. Mais l'approche qui sous-tend ce travail n'est-elle pas fonctionnaliste, au moins en partie ? Qu'extrait-on de la particularité d'une fourmi ?

De l'avis d'Edelman, le cerveau n'est pas réductible à un programme exécutable sur une machine ([Edelman, 1992]). En outre, il déclare : « *L'aptitude du système nerveux à effectuer une catégorisation perceptive des différents signaux pour la vue, le son, etc., et à les diviser en classes cohérentes sans code préalable est propre au cerveau, et les ordinateurs n'y parviennent pas* ». La raison principale qu'il invoque est qu'un programme est conçu en dehors de la réalité physique de la machine qui le fait fonctionner, même si des interactions avec le monde extérieur sont traitées dans ce programme. Cela signifie que deux machines différentes, possédant le même programme, traiteront le programme de la même façon ; en définitive, pour que celui-ci puisse s'exécuter à partir de « supports » différents (en particulier lorsque la machine est pourvue de capteurs ou d'effecteurs, cette différence, aussi minime soit-elle, existe), il faut avoir prévu l'ensemble des cas possibles au cœur du programme. Or, de son point de vue, en ce qui concerne le cerveau, le contenu (c'est-à-dire le programme) ne peut être dissocié du contenant (la réalité physique du corps). Pratiquement, nous constatons qu'un même programme réagit de manière différente sur deux robots distincts, possédant forcément des capteurs dont la réponse n'est pas tout à fait la même. Il faudrait donc que ce programme puisse s'adapter à la réalité physique du robot.

Certains faits témoignant de l'extraordinaire plasticité du cerveau tendent à contredire le fait qu'un « programme » y soit logé et que celui-ci « s'adapte » par apprentissage. Ainsi, il n'est pas exceptionnel que le cerveau récupère toutes ses facultés malgré la présence d'une lésion. La région touchée peut être relayée par d'autres zones cérébrales demeurées intactes ; il arrive même que ce relais soit pris par une région qui n'appartient pas au même hémisphère cérébral que celui où la lésion s'est produite. En particulier, cela a été observé chez des patients qui, après avoir perdu la motricité d'une main, l'ont retrouvée : leur main était désormais commandée par les deux hémisphères, alors que, normalement, l'hémisphère gauche pilote la main droite et réciproquement.

Une autre approche que la tentative de reproduction directe de parties de la topologie du cerveau existe. Il s'agit d'une tentative d'explication de l'organisation de cet organe au travers de lois ou de principes. Cette démarche est, par nature, explicative et non guidée par l'obtention immédiate de comportements simulés satisfaisants par rapport aux comportements réels. Ainsi, Edelman tente d'expliquer le fonctionnement et l'organisation neuronale par l'utilisation de la loi régissant la sélection naturelle, fondée par Darwin : sa théorie porte le nom de « théorie de la sélection des groupes neuronaux »([Edelman, 1992]). L'originalité de la démarche consiste à

appliquer une loi du vivant qui, à l'origine, était destinée à expliquer l'évolution d'entités sur de très longues périodes, à l'organisation de cellules nerveuses sur de très courts laps de temps. La démarche d'Edelman est donc bien plus qu'une volonté d'explication chimique ou organique d'un mécanisme : elle a pour vocation de trouver une formulation cohérente de ce mécanisme par rapport à une hypothèse fondatrice, c'est-à-dire l'application de la loi de sélection. Cela permet par conséquent de prévoir. La validation de la théorie est double : d'une part, il s'agit de vérifier que les prévisions correspondent bien à la réalité biologique ; d'autre part, il y a une possibilité de simulation sur ordinateur du phénomène biologique. Dans le cadre de la théorie d'Edelman, cela a été partiellement fait, sur des problèmes simples.

6.2.5 Réflexion à propos de la démarche fonctionnaliste

Le problème central que nous avons évoqué dans les paragraphes précédents est de spécifier un référent de l'intelligence. Le choix de celui-ci impose une voie de recherche particulière et implique des conjectures fortes à propos de la manière dont on aborde le problème.

Actuellement, la reproduction de capacités humaines intelligentes est un enjeu au niveau de la recherche, mais aussi au niveau commercial. Des applications fonctionnelles existent déjà tant au niveau de la reconnaissance vocale et de la compréhension du langage (utilisé pour des serveurs vocaux, par exemple) qu'au niveau d'expertises complexes. Des équipes de recherche progressent dans la mise au point d'humanoïdes (vision, marche, démarche « sociale », imitation artificielle d'émotions). Les programmes ainsi construits peuvent utiliser ou non des techniques d'apprentissage pour leur mise au point. Le point commun à tous ces travaux est la démarche d'*engineering*, guidée par le respect d'un cahier des charges, plutôt qu'une tentative de compréhension profonde des mécanismes humains qui sont imités. Cela ne signifie pas que les développements ne sont pas aidés par un savoir-faire biologique.

Nous souhaitons montrer, dans les lignes qui suivent, que la notion de progrès dans la compréhension de l'intelligence et de l'apprentissage en particulier n'est pas équivalente au fait d'obtenir une imitation très réussie d'un ensemble de comportements intelligents. Cette affirmation peut sembler triviale, voire superflue. Cependant, lorsqu'on analyse l'application de la démarche fonctionnaliste à l'imitation d'un comportement, on s'aperçoit qu'elle rentre en conflit avec la notion même d'adaptation.

Ainsi, si l'on adopte une vision fonctionaliste, on tente d'extraire des caractéristiques objectives d'un problème, indépendantes des propriétés physiques de la machine. Or, en robotique mobile, entre autres, celle-ci est en interaction permanente avec son environnement. Dans ce cas, l'idée fonctionaliste implique que le monde extérieur à la machine (l'environnement, mais aussi les capteurs et les moyens d'action) puisse être vu d'une manière objective (voir la figure 6.5).
L'analyse du problème global va permettre de dégager un certain nombre de fonctionnalités qu'il s'agira de reproduire séparément les unes des autres, selon le schéma donné par la figure 6.6. Avant de construire un comportement, on va établir une mesure objective qui permettra de conclure que celui-ci est bien reproduit s'il est « proche » de celui qui est désiré, au regard de cette mesure. La mesure créée devient le référent pour l'expérimentateur. Dès lors que le problème initial (complexe) est résumé simplement par une donnée (la mesure), on peut utiliser la démarche de résolution spécifiée par la figure 6.7. Néanmoins, celle-ci est loin de garantir qu'on ait compris l'essence du phénomène imité. L'exemple de l'étude du mouvement des corps célestes nous permet d'imager nos propos en fournissant un exemple d'analyse réussie d'un problème,

dont le modèle très compliqué donnait un bon résultat, mais dont l'approche était fausse[1]. Cet exemple est marquant car il montre que le résultat dans un cas particulier (prévoir l'évolution de la planète Mars) est quasiment le même, bien que la conception analytique s'oppose à la démarche synthétique, qui stipule *a priori* l'existence d'une loi à partir de laquelle des phénomènes observables émergent (voir les figures 6.9 et 6.8). Nous pensons que le processus d'adaptation devrait posséder une certaine généralité et, par conséquent, n'être pas tributaire d'un problème particulier ou d'une certaine classe de problèmes.

Nous souhaitons à présent aborder le point pour lequel la démarche analytique peut rentrer en conflit et nuire à l'utilisation d'un système adaptatif. Tout d'abord, lorsqu'on souhaite reproduire un comportement à partir d'une machine, une question doit logiquement être posée : *Est-il réaliste de créer un algorithme permettant de reproduire ce comportement ?*. Cela suppose que nous ayons des connaissances assez précises et suffisantes à propos du modèle de ce comportement pour que le résultat soit acceptable. Mais, « réaliste » signifie également que le temps de conception du programme n'est pas rédhibitoire et que celui-ci est maintenable. Si la réponse est affirmative, il est évident que le recours à une méthode d'apprentissage ne s'impose pas.

Cependant, dès qu'il s'agit de traiter des problèmes liés à l'interaction entre la machine et son environnement, il devient plus délicat de pouvoir produire un algorithme d'une manière directe (manque d'informations précises sur le modèle de l'interaction). Cela ne veut pas dire qu'il est

1. La cosmologie a longtemps été inspirée par les idées d'Aristote. L'une d'entre-elles était que le mouvement de tout corps céleste est circulaire. En effet, le cercle était associé à la perfection, donc à l'idée de divinité. Le ciel étant de domaine des dieux, les corps célestes devaient posséder ce mouvement circulaire. Une autre stipulait que la terre était le centre de l'Univers. Dans ce contexte, le référent est la comparaison des trajectoires observées avec les modèles qu'ils avaient élaborés. Or, en utilisant les hypothèses (fausses) d'Aristote, l'approximation de la trajectoire des planètes a donné lieu à des modèles épicycloïdaux, où le mouvement des planètes s'inscrit dans des épicycles s' appuyant sur des cercles qui demeurent centrés sur la Terre. Il est intéressant de constater qu'à partir d'hypothèses fausses (l'hypothèse géocentrique et l'utilisation de trajectoires circulaires imbriquées), les résultats obtenus ont été très satisfaisants. Par contre, les modèles étaient particuliers à chacune des planètes étudiées et étaient très compliqués (la trajectoire de Mars était très bien reproduite avec des modèles contenant jusqu'à une vingtaine d'épicycles imbriquées). À partir du modèle héliocentrique de Copernic, Kepler formule des lois concernant l'évolution des planètes. Cette découverte signifie plus qu'un changement d'hypothèse : le dogme d'Aristote tombe, ainsi que l'idée que la mécanique céleste n'est pas compréhensible intrinsèquement (par la découverte de lois physiques), mais simplement observable. Le référent n'est plus simplement une concordance la plus exacte possible du modèle et de la réalité dans un contexte précis (une planète précise) mais la concordance d'une loi avec les différentes observations (généralisation à un ensemble de planètes). Le mouvement des planètes n'est plus un ensemble de cas particuliers, car les trajectoires sont la manifestation de lois physiques.

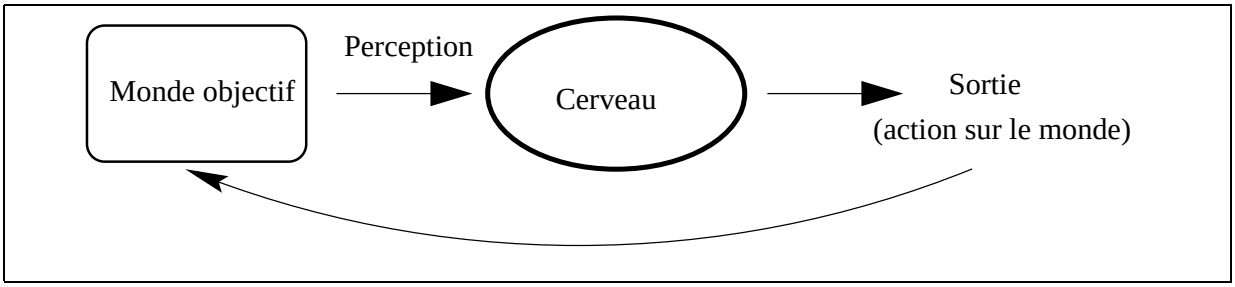


Fig. 6.5 – Aspect du fonctionnalisme : nécessité d'un monde pouvant être décrit objectivement

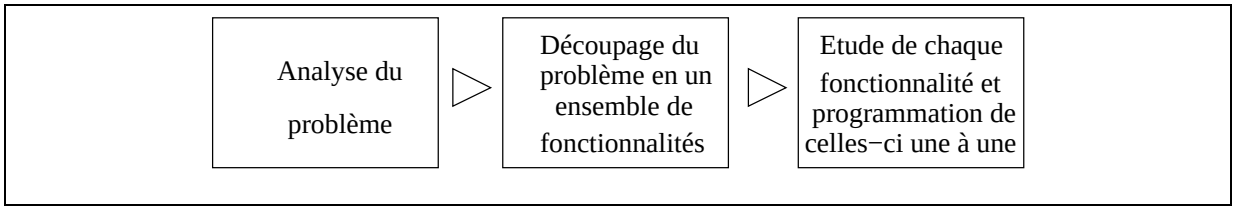


FIG. 6.6 – Démarche analytique globale de conception d'un problème complexe

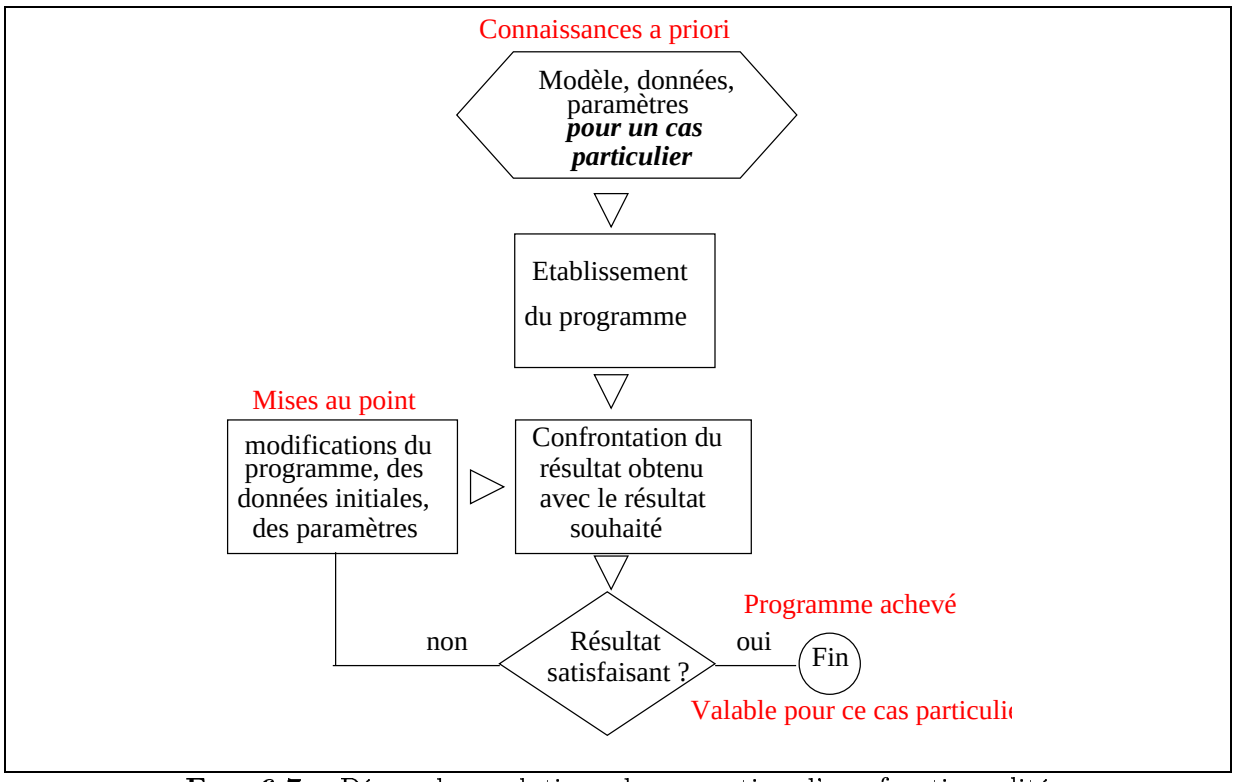


FIG. 6.7 – Démarche analytique de conception d'une fonctionnalité

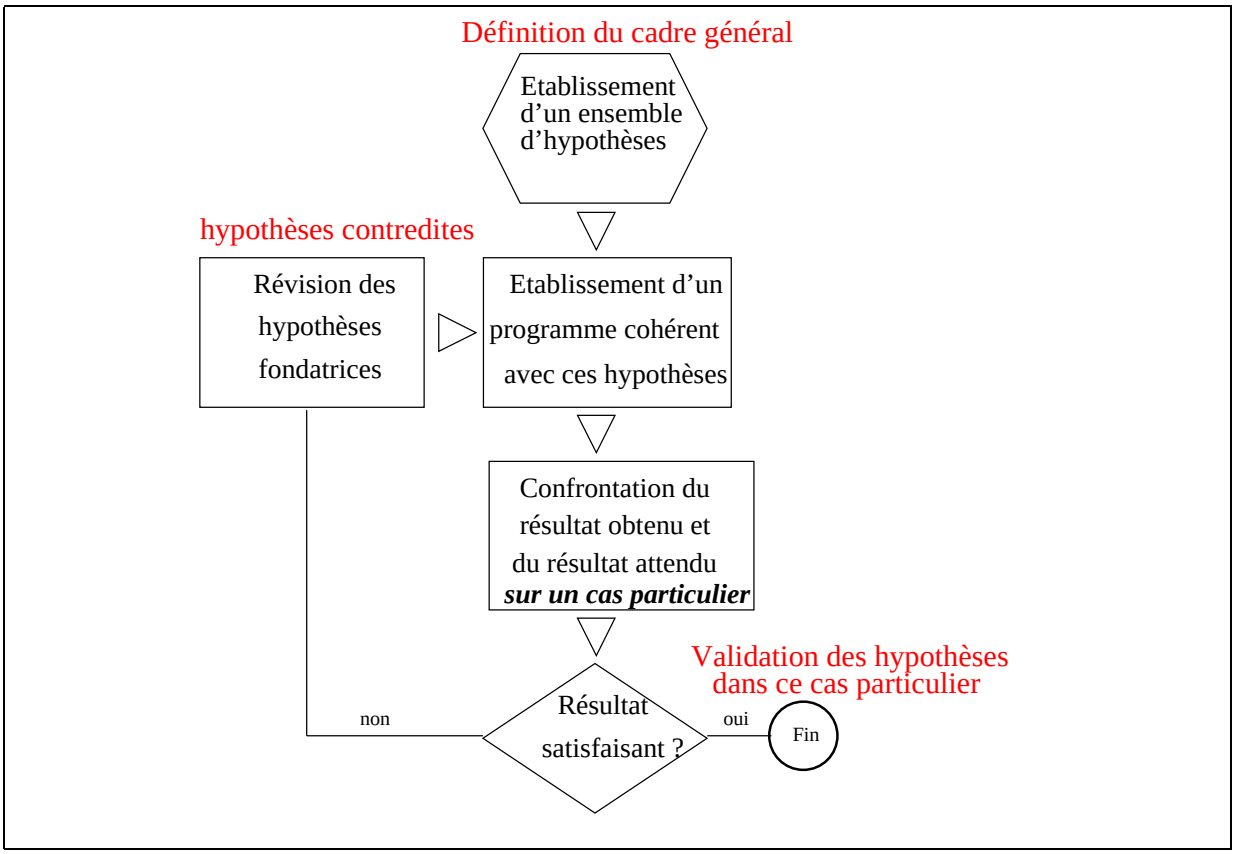


FIG. 6.8 – Démarche synthétique de conception d'une fonctionnalité

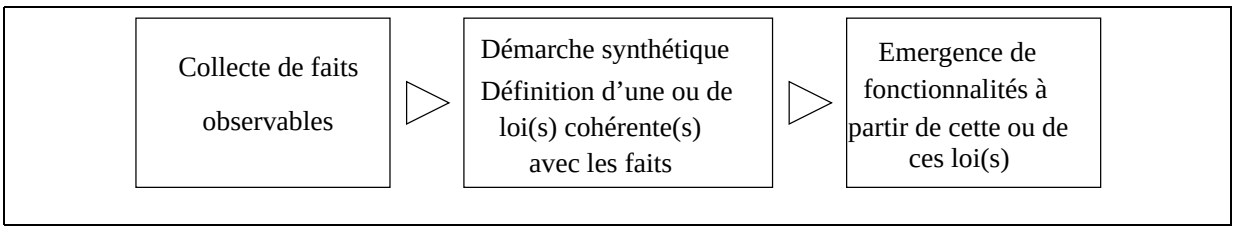


FIG. 6.9 – Démarche synthétique globale

impossible de réaliser « à la main » un tel algorithme, mais on suppose que le temps de conception serait très long et le programme obtenu très compliqué : cela ne serait donc pas « réaliste ». Dans ce cadre, on peut être intéressé par l'utilisation d'algorithmes d'apprentissage.

Cela ne signifie pas que l'ensemble du problème doit être appris en globalité. Ainsi, le travail de conception préliminaire consiste à distinguer les parties sujettes à l'apprentissage des autres pouvant être programmées *ad hoc* et à mettre en place l'interface entre les deux. Cela revient à répondre à ces deux questions : *Que savons nous (avec certitude) ? Quelle est la nature de ce qui doit être appris ?*, la deuxième question étant le corollaire de la première.

En fait, la démarche analytique, si elle est appliquée à la reproduction d'un processus mettant en jeu une interaction entre la machine et son environnement, va permettre de trouver des solutions, souvent approximatives, qui vont « coller » à un problème particulier. Or, quelle est la signification de l'adaptation, sinon la faculté de « coller » à un problème précis suffisamment bien, sans avoir de connaissances *a priori* sur celui-ci ? Par conséquent, dans le cas général, c'est davantage le concepteur du programme qui fait preuve d'adaptation au problème posé que le programme lui-même. Nous constatons en effet que le bon fonctionnement d'un algorithme d'apprentissage est corrélé à l'existence d'un contexte qui découle de l'analyse du problème initial.

Ainsi, en pratique, le point délicat d'un programme ayant recours à une technique d'apprentissage réside dans l'apport de connaissances *a priori* exactes, permettant de guider l'apprentissage : il s'agit d'un processus de contextualisation. Il est clair que dans le cas où ces données ou modèles sont trop approximatifs ou faux, ils vont orienter négativement le résultat final, sans que l'apprentissage lui-même soit la cause réelle de l'échec. Les algorithmes d'apprentissage sont créés en supposant l'existence d'un contexte précis, donc de connaissances *a priori* les plus exactes possible. Voici les principales :

- les paramètres internes et leur évolution au cours du temps, le cas échéant.
- les entrées et sorties et leur évolution au cours du temps, le cas échéant.
- une fonctionnalité permettant de juger de la qualité de l'apprentissage en cours (introduction d'une distance, d'une fonction de coût) et induisant la conduite vers un objectif précis.

D'autres causes de contextualisation peuvent exister dont, entre autres :

- des connaissances sur la particularité de l'environnement, composé d'un ensemble fini de symboles perceptifs (voir la discussion à ce propos au paragraphe 6.2.3)
- des connaissances à propos du fonctionnement des moyens de perception et d'action de la machine (utilisation de modèles)

Donc, pour pouvoir utiliser un algorithme d'apprentissage, l'expérimentateur est censé connaître (ou découvrir par tâtonnements) des éléments qui sont au cœur même du processus d'interaction entre la machine et son environnement. Il peut s'aider des propriétés ou du modèle de l'environnement de la machine. Mais toutes ces connaissances sont bien souvent approximatives et fondent un contexte erroné. L'expérience montre qu'il faut très souvent « travailler » autour de l'algorithme d'apprentissage pour que le résultat soit correct ; et c'est particulièrement le cas si on utilise des techniques d'apprentissage par renforcement (voir [Bersini et Gorrini, 1996] en ce qui concerne la difficulté à trouver, dans certains cas, les « bonnes » valeurs des paramètres internes). Nous pensons que l'enjeu principal d'un système adaptatif est de pouvoir permettre à la machine d'apprendre avant tout à utiliser ses sens et ses moyens d'action, l'obtention du

comportement désiré venant ensuite aisément : par conséquent, nous croyons qu'un ajout de connaissances orientant *a priori* la perception n'est pas souhaitable (car ces données ne sont pas connues exactement) et que le processus de perception doit lui-même subir un apprentissage, qui permettra l'acquisition correcte du comportement désiré. Ce processus de perception, une fois maîtrisé, contextualise correctement le problème principal, c'est-à-dire l'acquisition d'une nouvelle compétence. Notre point de vue est que l'apprentissage n'est pas un problème « difficile » si son contexte est parfaitement défini. Et c'est ce dernier point qui pose réellement problème. Ainsi, dans le cadre de l'apprentissage de réseaux neuronaux multicouches, le choix d'une base d'apprentissage « brute » représentative du phénomène est nécessaire, mais le pré-traitement des données de cette base et/ou la réduction de la dimensionnalité du problème peut être une aide considérable à l'apprentissage : ces deux éléments sont contextuels.

6.2.6 Conclusion - Liens avec la biologie et avec le concept de la machine de Turing

Commençons par rappeler les idées importantes issues de cette section. Dans un second paragraphe, nous expliquerons comment notre approche se positionne.

Nous pensons que l'objet de l'apprentissage doit être centré en particulier sur la faculté de perception elle-même (savoir repérer l'information utile au bon moment, dans le cadre d'un objectif déterminé). A ce titre, les connaissances ajoutées *a priori*, concernant le processus d'interaction entre la machine et son environnement amoindrissent la possibilité d'adaptation du système en figeant certains traits de la perception. Or, si ces données ne sont pas suffisamment exactes, le résultat de l'apprentissage en est affecté, sans que celui-ci en soit la cause de l'échec. Nous pensons que l'analyse d'un problème conduisant à l'élaboration d'un comportement intelligent doit permettre de discerner les frontières entre des caractères certains et exacts, qui seront abordés dans une logique fonctionnaliste (avec une démarche de programmation « classique ») et des caractères incertains, qui seront appris, et dont les processus de perception font intégralement partie. Pour une machine et un environnement donnés, l'apprentissage doit permettre, par l'expérience, d'appréhender la réalité physique de leur interaction (particularité des capteurs et de l'environnement). Il remplace donc une analyse *a priori* du problème posé par la singularité de cette interaction (modélisation des capteurs et de l'environnement), sachant que cette dernière ne peut pas être faite, dans bien des cas, d'une manière suffisamment exacte ou exhaustive. En particulier, cela signifie qu'une programmation *ad hoc* du phénomène d'interaction est inenvisageable dans le cas général, car elle nécessite que l'intégralité du phénomène puisse être transcrite en un ensemble d'instructions exactes, fonctionnant sur une machine dont les lois d'évolution sont figées. Enfin, nous pensons que les mécanismes qui permettent de découvrir cette interaction ne sont pas contextuels : en particulier, ils ne dépendent pas de la nature du comportement à imiter, ni de la nature des signaux perceptifs.

Notre approche s'écarte de la démarche fonctionnaliste : nous n'essayons pas de reproduire ou d'imiter un phénomène particulier, ni une structure cérébrale. Par conséquent, notre démarche n'est pas d'utiliser des observations pour construire directement un modèle : il s'agit d'utiliser des observations pour supposer l'existence de propriétés qui, elles, suggèrent l'existence d'un modèle précis. Nous nous inspirons à la fois de l'approche biologique, en utilisant le principe de sélection, mis en avant par Edelman entre autres, mais sans nous écarter beaucoup du modèle de la machine de Turing. En effet, nous rappelons au lecteur que l'AP doit conduire à la création d'un ensemble d'états vérifiant la propriété (P_ϵ) : en d'autres termes, l'objectif de l'AP est de

construire un ensemble d'états pour lesquels on peut prédire avec une grande certitude l'état futur du système si on connaît son état actuel et la commande qui est exécutée. Nous rappelons ici que la différence avec le concept de la machine de Turing est qu'il n'existe pas de déterminisme absolu associé à la propriété (P_ϵ) : la prédiction est théoriquement incertaine, mais la probabilité d'occurrence d'une mauvaise prédiction est tellement faible que celle-ci ne se produit jamais en réalité, pour une durée d'expérience « raisonnable ».

6.3 Une certaine notion de la réalité

6.3.1 Introduction

Les choix que nous avons effectués dans notre démarche de recherche, qui sont une conséquence directe des hypothèses fondatrices de notre travail, impliquent une conception du monde (notion de "réalité") qui est éloignée des domaines scientifiques dans lesquels notre travail est censé s'inscrire (apprentissage automatique, traitement du signal, ingénierie en général). Pourtant, en essayant de trouver une approche similaire "quelque part" en physique, nous nous sommes aperçu, avec étonnement, que les pré-supposés de notre démarche coïncident correctement avec ceux exprimés en micro-physique.

Bien entendu, il ne s'agit pas d'effectuer une comparaison entre le formalisme utilisé en mécanique quantique et ce travail : nous en serions incapable et cela n'aurait, à notre avis, aucun sens. Nous sommes bien conscients qu'il faut bien se garder d'utiliser des analogies à tout prix, puisque cela conduit très rapidement à des « divagations » de l'esprit.

Nous souhaiterions revenir en particulier sur la notion d'observation (évoquée dans une section introductive de notre document).

6.3.2 Lien entre information perceptive et observation

Lorsque nous avons introduit la notion *d'information perceptive*, nous avons signalé que ce terme n'avait pas de rapport direct avec l'acception du terme *information* en traitement du signal : dans notre cas, l'information perceptive n'est pas portée uniquement par le signal, mais elle est le résultat du processus de catégorisation, utilisant comme entrées le signal et la mémoire du système. Ainsi, l'information perceptive possède un rapport plus direct avec la notion de *détection* d'un objet ou d'une propriété physique, rendue possible par un dispositif expérimental précis. Nous relierons cette détection à la notion plus générique d'observation.

Nous avons supposé les faits suivants :

1. l'information perceptive est le résultat d'un processus dynamique (processus de sélection) initié à partir d'un a priori (processus d'anticipation)
2. l'a priori est constitué d'un ensemble d'hypothèses, fournissant potentiellement autant d'informations perceptives possibles (celles-ci sont déterminées a priori grâce au processus d'anticipation)
3. l'information perceptive proprement dite (c'est-à-dire ce qui est effectivement observé) est constituée de l'ensemble restreint des hypothèses déterminées par le processus d'anticipation qui se sont révélées hautement probables (par confrontation avec la "réalité") : l'information perceptive ne peut pas être séparée du fait que l'observateur (le système utilisant le

processus de catégorisation) est (quasi-)certain d'avoir observé quelque chose[1]. Dans cette condition, il se peut qu'aucune information perceptive ne soit produite. Ce cas se produit si la totalité des hypothèses est rejetée. Ce rejet est dû au fait que l'ensemble des hypothèses initiales est inadapté à cette "réalité".

4. les hypothèses possèdent en elle-même la durée nécessaire à leur validation (valeur de h). Celle-ci est déterminée à partir de notre postulat de rareté de l'information perceptive.

A partir du point 4, on peut noter certaines remarques importantes :

- la probabilité considérée dans 4 ne peut pas être nulle suivant notre formalisation, puisqu'elle induirait une durée de validation infinie des hypothèses.
- la localisation du signal s'effectue dans l'espace et le temps : chacune des hypothèses est identifiée à une hypothèse possédant une trajectoire déterminée, sur une certaine durée.

6.3.3 Notion de la réalité, dérivée de l'information perceptive

Il faut se souvenir que le postulat fondateur de notre approche est l'existence de principe(s) ou de loi(s) permettant d'unifier le processus de catégorisation. Une telle conjecture implique obligatoirement des pré-supposés sur la nature de la "réalité". Il est indispensable de se demander ce que ces pré-supposés disent à propos de celle-ci. En particulier, existe-t-il des liens avec une des nombreuses définitions de la "réalité" ?
Une vision "classique" de la réalité consiste à penser qu'elle est une somme de phénomènes existant indépendamment de l'observateur. Or, notre exposé traduit une toute autre vision qui met en jeu deux réalités et qui est celle qu'expose d'Espagnat, à propos de la réalité quantique :

- une *réalité intentionnelle* dans laquelle l'information perceptive dépend des caractéristiques internes de l'observateur (processus d'anticipation). Par "intentionnelle", nous entendons qu'elle est guidée par un a priori, qui dépend d'un objectif - ou d'une intention - précis.
- une *réalité indépendante*[2] du système (qui fournit les valeurs du signal dans notre cas).

Les deux réalités sont liées par l'application du paradoxe opposant la rareté théorique des observations (informations perceptives) et la fréquence concrète de ces dernières. Dans ce système de pensée, l'observation (ou l'information perceptive) fait partie de la réalité "intentionnelle" ; son caractère "restreint" est engendré par les caractéristiques propres de l'observateur (intention, moyens de perception), ainsi que par les contraintes agissant sur le système (contraintes CO et CU) et non pas par la réalité indépendante. Par "restreint", nous entendons que seul un petit nombre d'hypothèses est validé, en rapport à l'infinité des hypothèses d'évolution possibles. Cela signifie simplement qu'on va observer ce qu'on cherche (intentionnalité), si toutefois le moyen d'observation (processus d'anticipation) est adéquat (l'apprentissage perceptif a eu lieu). Cela signifie également que deux entités différentes (par leurs moyens de perception ou par leur expérience) observent nécessairement différemment.

Il est important de comprendre la nature particulière que l'observation prend ici : elle représente la "concrétisation" (par sélection) d'un univers des possibles (c'est à dire de la mémoire du système), qui ne fait pas partie de la réalité intentionnelle (elle est à l'entrée du processus de catégorisation). L'observation possède *de facto* un caractère de localité (dans l'espace et dans le temps, donné par la variable h). Nous rappelons à ce sujet que le formalisme auquel nous aboutissons interdit qu'il existe une observation infiniment précise, ou une information liée à un instant

1. Nous faisons implicitement le rapprochement entre le sentiment de certitude, que l'être humain peut ressentir sur la réalité de ce qu'il perçoit, et la fiabilité de l'information perceptive, telle que nous la décrivons dans ce document de thèse. 2. Le terme « réalité indépendante » est emprunté à d'Espagnat.

donné (durée infiniment petite) : l'observation est associée à une durée non nulle dépendant de h. De même, nous supposons que la durée associée à une observation possède une borne supérieure (le contraire devrait aboutir à une contradiction, mais nous ne l'avons pas montré). En d'autres termes, le fait même d'obtenir une information perceptive suppose qu'elle s'inscrive dans une période de temps limitée et un morceau de l'espace lui aussi limité : il s'agit de propriétés **intrinsèques** de l'observation et non de limitations qu'on pourrait dépasser en améliorant la qualité des organes de perception, par exemple. Ainsi, dans notre système, une observation idéale n'a pas de sens : même si on pouvait imaginer un mode de perception infiniment précis, il faudrait pouvoir imaginer également une mémoire comportant un trop grand nombre d'hypothèses, qui contredirait les contraintes CO et CU.

Voici une citation de d'Espagnat, décrivant la méthode des définitions partielles inspirée par Carnap. Il s'agit de montrer la relation qui existe avec notre raisonnement sur la nature du processus de catégorisation.

> *Selon la méthode des définitions partielles, un système quantique S ne peut avoir les propriétés A que lorsque le dispositif expérimental permet une mesure de A. En cas contraire l'assertion "S a la propriété A" est "pire que fausse" car elle est tout simplement dénuée de sens. [...] Dans "l'esprit de Copenhague" les conditions expérimentales définissant les types possibles de prédiction sont essentielles. En d'autres termes la spécification du dispositif expérimental destiné à interagir avec le système étudié doit être considérée avant même que l'on puisse parler des propriétés de ce dernier ; et c'est elle seule qui détermine celles de ces propriétés dont il y a sens de parler."*

Il est également important de souligner que l'information perceptive est associée à un "bout" d'espace et à un "bout" de temps". Dans notre cas, on ne peut pas dire que ces "bouts" d'espace et de temps possèdent une réalité en dehors de la mémoire elle-même. Cela paraît très choquant à première vue, car on imagine habituellement l'espace comme un contenant pouvant recevoir des objets à l'intérieur (à l'image d'un aquarium). De même pour le temps. Ces contenants auraient alors naturellement une existence propre, indépendante de la chose (le contenu). De même, on y associera naturellement une norme. Or, une caractéristique de notre démarche est de refuser d'utiliser une norme quelconque afin de comparer deux informations perceptives. Une autre caractéristique est de supposer que les processus d'anticipation et de sélection ne s'appliquent qu'à des signaux mono-dimensionnels. Il est supposé que ces processus d'appliquent "simultanément" à une collection de signaux, et qu'une "corrélation" entre les évolutions "simultanées" de plusieurs signaux donnent l'illusion d'un espace à deux, trois ... dimensions. Ainsi, la dimensionalité (apparente) de l'espace fait également partie de la réalité intentionnelle. Or, la spécification de cette "corrélation" et de ce terme "simultanée" reste entièrement à faire. Pour rester cohérent, il est indispensable qu'elle réside sur l'utilisation de contraintes, qui est le fil conducteur du travail.

En résumé, la notion de réalité présentée ici est en contradiction avec la vision classique de la réalité sur beaucoup de points, à savoir :

- l'observation ne peut être dissociée de l'observateur, mais procède également de l'existence d'une réalité indépendante de l'observateur.
- l'espace et le temps sont indissociables de l'objet se trouvant dans l'espace pendant une certaine durée.

- l'espace et le temps ne sont pas sécables en une ensemble de morceaux aussi petits qu'on le désire, puisqu'ils sont liés à une observation particulière, associée à un bout d'espace et un bout de temps de mesure non nulle selon notre formalisme.
- la dimensionnalité de l'espace fait partie intégrante de l'observation et n'est pas une constante isolée de l'observateur.
- la réalité indépendante ne peut pas, par définition, être décrite (comme un ensemble de phénomènes).
- la réalité intentionnelle est dépendante a priori d'un observateur particulier et n'a pas de caractère universel.

Le problème qu'il nous reste à traiter pour achever notre processus de catégorisation est celui de la causalité, c'est-à-dire de l'existence d'une concordance temporelle entre différents signaux (externes ou internes). Nous pensons que la causalité fait partie de la réalité intentionnelle, donc qu'elle est engendrée par le processus de catégorisation, sur une échelle de temps supérieure à la durée de validation d'une hypothèse. Il faut bien comprendre que le délai entre la demande d'information perceptive et la concrétisation de celle-ci correspond à une durée insécable pour le système : aucune information ne peut être disponible au bout d'un temps plus court. Ainsi, le « bout de temps » spécifié par la valeur de h correspond à la notion d'instant : des informations perceptives concrétisées sur une durée inférieure à cette durée minimum sont perçues comme étant simultanées. La causalité apparaît pour des durées supérieures : Il faut pouvoir "lier" ces "bouts de temps" et ces "bouts d'espace" entre eux, pour faire naître la notion de continuité perceptive (O'Regan entre autres a bien montré qu'il s'agit d'une illusion) et celle d'une causalité perception/action.

Nous terminerons par une dernière citation de d'Espagnat, qui exprime ce que nous venons de dire bien mieux que nous :

> *[...] je tends à voir dans la causalité (au sens étroit, technique du terme) une structure de l'entendement humain, autrement dit un cadre a priori dans lequel l'esprit se trouve contraint, de par sa nature même, d'insérer toute son expérience, et non pas une structure de la réalité indépendante. En cela, je ne fais au surplus qu'expliciter une attitude qui est commune à la majorité des physiciens théoriciens contemporains et que ceux-ci fondent sur des caractéristiques importantes de leur théorie générale. [...] je trouve plausible l'idée de Kant selon laquelle espace et temps pourraient bien n'être eux aussi que des formes a priori de l'esprit humain, ce qui signifierait que la réalité indépendante n'est ni insérée dans ni composée de l'espace-temps."*

Références

Bain, A. (1873). *Mind and Body. The Theories of Their Relation.* Henry King, London.

Bersini, H. and Gorrini, V. (1996). Three connectionist implementations of dynamic programming for optimal control : A preliminary comparative analysis. In *Workshop on Neural Networks for Identification and Control in Robotics.*

d'Espagnat, B. (1985). *Une incertaine réalité.* Gauthier-Villars.

d'Espagnat, B. (1994). *Le réel voilé.* Fayard.

Edelman, G. (1992). *Bright Air, Brilliant Fire : On the Matter of Mind.* Basic Books, New York.

Hebb, D. (1949). *The Organization of Behavior.* John Wiley & Sons, New York.

Hilbert, D. et Ackermann, W. (1928). *Grundzuge der Theoretischen Logik.* Springer, Berlin.

Hsu, F., Anantharaman, T., Campbell, M., and Nowatzyk, A. (1990). A grandmaster chess machine. *Scientific American*, 263(4) :11–50.

James, W. (1890). *Principles of Psychology.* Henry Holt, New York.

Kohonen, T. (2001). *Self-Organizing Maps*, volume 30. Springer Series in Information Sciences, Berlin.

Lecerf, C. (1997). *Une leçon de piano ou la double boucle de l'apprentissage cognitif*, volume 3. Travaux et Documents, Université Paris 8 Vincennes-Saint-Denis.

McCulloch, W. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5 :115–137.

Pitrat, J. (1990). *Métaconnaissance - Futur de l'intelligence artificielle.* Hermès.

Pradel, G. and Barret, C. (1998). Environment recognition in mobile robotics by means of neural networks. *Journal Européen des Systèmes Automatisés*, 32 :939–963.

Revel, A. (1997). *Contrôle d'un robot mobile autonome par approche neuromimétique.* Thèse de doctorat, Université de Cergy-Pontoise.

Rosenblatt, F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65 :386–408.

Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning internal representations by error propagation. *Nature*, 323 :533–536.

Shortliffe, E. and Buchanan, B. (1975). A model of inexact reasoning in medicine. *Mathematical Biosciences*, 23 :351–379.

Turing, A. (1936). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42(2) :230–265.

Turing, A. (1950). Computing machinery and intelligence. *Mind*, 59 :433–460.

Weizenbaum, J. (1976). *Computer Power and Human Reason.* W.H. Freeman.