HAL
open science

# Reconnaissance de séquences d'états par le Modèle des Croyances Transférables. Application à l'analyse de vidéos d'athlétisme.

Emmanuel Ramasso

▶ **To cite this version:**

Emmanuel Ramasso. Reconnaissance de séquences d'états par le Modèle des Croyances Transférables. Application à l'analyse de vidéos d'athlétisme.. domain_stic.inge. Université Joseph-Fourier - Grenoble I, 2007. Français. NNT : . tel-00260770

**HAL Id: tel-00260770**

**https://theses.hal.science/tel-00260770**

Submitted on 5 Mar 2008

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

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

Université Joseph Fourier

École doctorale EEATS

# Reconnaissance de séquences d'états par le Modèle des Croyances Transférables Application à l'analyse de vidéos d'athlétisme

## THÈSE

Soutenue publiquement le 5 décembre 2007 pour l'obtention du titre de

**Docteur de l'Université Joseph Fourier de Grenoble**

spécialité

Electronique, Electrotechnique, Automatique et Traitement du Signal

par

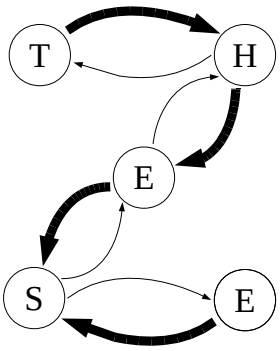
**Emmanuel RAMASSO**

| | | |
|---|---|---|
| Directeurs de thèse : | Denis PELLERIN | Professeur, Université Joseph Fourier |
| | Michèle ROMBAUT | Professeur, Université Joseph Fourier |
| Rapporteurs : | Thierry DENŒUX | Professeur, Université Technologique de Compiègne |
| | Patrick LAMBERT | Professeur, Université de Savoie |
| Présidente : | Catherine BERRUT | Professeur, Université Joseph Fourier |
| Examinateur : | Jean–Michel JOLION | Professeur, Université de Lyon |

**Laboratoire Grenoble Images Parole Signal Automatique (GIPSA-lab)**

**Département Images et Signal**
**Equipe G$\pi$G, 2004-2007**

# Reconnaissance de séquences d'états par le Modèle des Croyances Transférables Application à l'analyse de vidéos d'athlétisme




© Emmanuel Ramasso

e_ramasso@yahoo.fr

GIPSA-lab
Département Images et Signal
Grenoble, France

# Remerciements

Les travaux présentés dans ce manuscrit de thèse ont été financés par une bourse du Ministère de l'Education Nationale, de la Recherche et de la Technologie (MENRT).

J'adresse mes sincères remerciements aux rapporteurs de la thèse, Thierry Denœux et Patrick Lambert, dont les remarques et suggestions m'ont permis d'améliorer la qualité de ce manuscrit. Je remercie par ailleurs Jean-Michel Jolion et Catherine Berrut d'avoir accepté d'être dans mon jury en tant qu'examinateur et présidente respectivement.

Mes remerciements s'adressent tout naturellement à mes deux directeurs de thèse, Michèle Rombaut et Denis Pellerin pour tout ce qu'ils m'ont apporté tout au long de ces trois années. Je remercie Michèle pour m'avoir *évangélisé* aux fonctions de croyance ainsi que Denis pour m'avoir soutenu dans mes choix de recherche.

Je remercie J.-M. Chassery pour m'avoir accueilli au sein du Laboratoire *Grenoble Image Parole Signal Automatique* (GIPSA-lab) ainsi que et C. Jutten pour son accueil au *département Image et Signal* (DIS).

J'exprime toute ma reconnaissance aux services administratif, technique et informatique du GIPSA-lab ainsi qu'à l'Université Joseph Fourier et plus particulièrement à Marino, Jean-Marc, Nadia, Hervé et Laurent.

Que l'ensemble de mes collègues de travail du GIPSA-lab / DIS soient remerciés pour leur accueil et nos échanges et en particulier Mickaël, Brice, Nicolas, Rosilde, Nathalie, Pierre-Yves et Anne. Je remercie de même David Mercier et Benjamin Quost du laboratoire Heudiasyc pour nos discussions enrichissantes concernant les fonctions de croyance, et une pensée particulière va à Patrick Vannoorenberghe.

Je remercie le réseau d'Excellence Européen SIMILAR au sein duquel j'ai entretenu des relations très enrichissantes et fructueuses avec Costas Panagiotakis et Giorgos Tziritas de l'Université de Crète. Je remercie en particulier Costas pour ses échanges d'informations tout au long de ces trois années, pour sa bonne humeur et pour son amitié. Une pensée aussi pour Maria.

Enfin, je remercie ma femme, Sylvie, pour son soutien et sa patience ainsi que ma famille et ma belle-famille qui ont grandement contribué à la réalisation de cette thèse.

# Table des matières


**1 Introduction** **11**
1.1 Contexte et motivation . . . . . 11
1.2 Méthodologies pour la reconnaissance de séquences . . . . . 12
1.2.1 Méthodes symboliques . . . . . 12
1.2.2 Méthodes statistiques . . . . . 13
1.3 Contribution . . . . . 15
1.4 Organisation du mémoire . . . . . 15

**2 TBM : représentation, fusion, décision** **17**
2.1 Outils de base . . . . . 18
2.1.1 Présentation des concepts sur un cas d'école : le lancé de dé . . . . . 19
2.1.2 Formalisme des fonctions de croyance dans le TBM . . . . . 20
2.1.3 Le principe du minimum d'information . . . . . 23
2.1.4 Règles de combinaison de base . . . . . 24
2.1.5 Prise de décision à partir de fonctions de croyance . . . . . 27
2.1.6 Gestion du conflit . . . . . 28
2.2 Opérations avancées . . . . . 29
2.2.1 Le Théorème de Bayes Généralisé . . . . . 29
2.2.2 Opérations sur les cadres de discernement . . . . . 37
2.2.3 Déconditionnement . . . . . 40
2.2.4 Le PMI pour transformer une distribution de probabilités en BBA . . . . . 43
2.2.5 Combinaison de sources non distinctes . . . . . 45
2.2.6 Affaiblissement généralisé dit *contextuel* . . . . . 48
2.3 Conclusion . . . . . 48

**3 Filtrage de fonctions de croyance** **51**
3.1 Introduction . . . . . 51
3.2 Filtrage crédibiliste avec modèles à sauts . . . . . 53
3.2.1 Définition des concepts utilisés . . . . . 54
3.2.2 Les concepts sur un exemple : segmentation de vidéos . . . . . 54
3.3 Description du filtre temporel crédibiliste . . . . . 56
3.3.1 Principe du filtre . . . . . 56
3.3.2 Prédiction . . . . . 57
3.3.3 Mise à jour . . . . . 59
3.3.4 Détection de changement d'état . . . . . 60

3.3.5 Sortie du filtre : gestion du conflit . . . 61
3.4 Critère de qualité locale . . . 63
3.5 Estimation des paramètres du filtre . . . 64
3.5.1 Données d'apprentissage . . . 64
3.5.2 Seuil d'arrêt $\mathbb{T}_s$ et coefficient d'oubli $\lambda$ . . . 65
3.5.3 Coefficients des modèles $\mathcal{T}$ et $\mathcal{F}$ . . . 67
3.5.4 Initialisation du filtre . . . 69
3.6 Amélioration du filtre . . . 69
3.6.1 Changement de modèle crédibiliste . . . 70
3.6.2 Généralisation du filtre à N états . . . 70
3.7 Conclusion . . . 72

**4 Reconnaissance déterministe de séquences d'états par machine à états finis crédibilistes 75**
4.1 Introduction . . . 75
4.2 Approche crédibiliste pour la reconnaissance de séquences . . . 77
4.2.1 Principe de l'ordonnanceur . . . 77
4.2.2 Définition des concepts . . . 78
4.3 Mécanismes de préemption et de forçage pour synchroniser les états . . . 81
4.3.1 Gestion des états superposés : *préemption* . . . 83
4.3.2 Gestion des états disjoints : *forçage*, et détection de fausses alarmes . . . 84
4.4 Inférence et classe de rejet . . . 87
4.5 Conclusion . . . 90

**5 Reconnaissance non-déterministe de séquences d'états par modèles de Markov cachés crédibilistes 93**
5.1 Introduction . . . 93
5.2 HMM probabiliste . . . 93
5.2.1 Modèle de séquence et observations . . . 94
5.2.2 Mécanismes de base : propagations *forward* et *backward* . . . 96
5.2.3 Estimation de la probabilité sur chaque état et filtrage . . . 98
5.2.4 Recherche de la meilleure séquence d'états . . . 99
5.2.5 Classification d'une séquence d'observations . . . 102
5.2.6 Apprentissage des paramètres . . . 104
5.2.7 Pourquoi étendre les HMM aux fonctions de croyance ? . . . 106
5.3 HMM crédibiliste . . . 107
5.3.1 Introduction . . . 108
5.3.2 Mécanismes de base . . . 108
5.3.3 Croyances sur chaque état et filtrage . . . 111
5.3.4 Recherche de la meilleure séquence d'états . . . 113
5.3.5 Classification de séquence d'observations . . . 119
5.3.6 Vers l'apprentissage des paramètres . . . 123
5.4 Conclusion . . . 126

**6 Reconnaissance d'actions et d'activités humaines dans des vidéos d'athlétisme 129**
6.1 Introduction . . . 129
6.1.1 Analyse des vidéos de sports . . . 130
6.1.2 Analyse de mouvements humains . . . 131
6.1.3 Contribution . . . 133
6.2 Contexte expérimental . . . 134
6.2.1 Architecture du système proposé . . . 134
6.2.2 Description de la base de vidéos . . . 134
6.3 Extraction . . . 136
6.3.1 Estimation du mouvement de caméra . . . 136
6.3.2 Catégorisation de la vidéo : sport collectif vs. sport individuel . . . 141
6.3.3 Suivi de points . . . 143
6.3.4 Les caractéristiques de mouvement utilisées . . . 144
6.4 Modélisation . . . 148
6.4.1 Problème posé . . . 149
6.4.2 Vraisemblance et Théorème de Bayes Généralisé . . . 150
6.4.3 Le modèle de distance . . . 151
6.4.4 Modélisation *experte* . . . 152
6.5 Evaluation . . . 155
6.5.1 Réglages des algorithmes . . . 156
6.5.2 Reconnaissance des actions dans les activités avec la méthode symbolique . . . 156
6.5.3 Reconnaissance des actions dans les activités avec des méthodes statistiques . . . 161
6.5.4 Classification de séquences . . . 167
6.6 Conclusion . . . 168

**Conclusion générale et perspectives 171**

**Annexes 175**

**Bibliographie 183**

**Liste des publications 199**

**Liste des tableaux 205**

**Table des figures 212**

# Chapitre 1

# Introduction

## 1.1 Contexte et motivation

La détection, la modélisation ou encore la reconnaissance représentent des problématiques majeures dans les communautés du Traitement du Signal, du Traitement d'Image et de la Vision par Ordinateur. A partir de l'observation d'un phénomène, on cherche à *extraire*, *modéliser* et *combiner* des informations diverses pour améliorer la *prise de décision* [Blo05]. Ces quatre étapes constituent les éléments de base d'un système de *fusion d'information* [Val01] et chacune de ces étapes est une problématique à part entière dont les solutions dépendent de l'application.

Dans de nombreux domaines, le résultat est à destination d'un opérateur humain et donc il est nécessaire que les informations qui lui sont transmises soient d'un haut niveau d'interprétation. C'est le cas, par exemple, pour la gestion de bases de données vidéos, ou bien pour une application de diagnostic.

Dans le cadre de la gestion des bases de vidéos, l'utilisateur émet des requêtes symboliques en utilisant des concepts de haut niveau sémantique. Le système doit être capable d'interpréter ces requêtes et de satisfaire l'utilisateur en retournant les vidéos qui l'intéressent. Si celui-ci n'est pas satisfait, le système peut prendre en compte les remarques (*feedback*) afin d'améliorer sa performance. La différence de niveau conceptuel entre la machine et l'utilisateur est surnommée *fossé sémantique* [LHGT04] et l'étape de modélisation permet de combler ce fossé. La modélisation est d'autant plus aisée que les données bas niveau extraites, aussi appelées *caractéristiques*, sont interprétables. Lorsque les caractéristiques ne le sont pas, des outils statistiques, généralement assimilés à des *boîtes noires*, sont utilisés. Ces outils sont des algorithmes d'apprentissage qui estiment des modèles en minimisant ou en maximisant un critère afin que ceux-ci correspondent au mieux aux données. L'avantage de ces méthodes statistiques est de pouvoir modéliser des problèmes complexes lorsque les modèles sont inconnus, par exemple pour l'analyse de mouvements humains dans les vidéos [BR04, MHK06, JLBA07], la reconnaissance de la parole [BB02, VH06] et l'analyse de données textuelles [Kel07]. Le problème avec ces méthodes est que, d'une part, les modèles générés ne sont généralement pas interprétables et, d'autre part, il est difficile, même pour des experts, de les mettre à jour sans passer par une nouvelle étape d'apprentissage qui peut être fastidieuse.

Dans le domaine du diagnostic, par exemple pour des applications militaires [XHS96, MB03, RS05b], médicales [Sme98, RZ02, AAG02, Str04] ou industrielles [Val01, Val03], les enjeux sont importants au point de laisser la prise de décision à un opérateur humain qui prendra le risque de

la décision à sa charge. Comme dans l'exemple précédent, les informations fournies à l'opérateur doivent être interprétables afin qu'il puisse raisonner et élaborer un jugement. Ces informations proviennent généralement de la sortie des étapes de modélisation (avant combinaison) et de combinaison (avant transformation pour la prise décision) dont la nature dépend du formalisme utilisé pour gérer les imperfections des données d'apprentissage. Outre la théorie des probabilités largement exploitée depuis des décennies, la théorie des possibilités et celle des fonctions de croyance [KW99], plus récentes, offrent des outils puissants de gestion de l'incertitude, de l'imprécision et du conflit sur ces données. Les méthodes de modélisation et de combinaison possibilistes et crédibilistes facilitent l'interprétation et l'intégration de connaissances [Val03, MB03, RS05b]. La prise de décision dans le cadre de ces théories est généralement basée sur une *distribution de probabilités* obtenue par défuzzification dans le cas possibiliste [VMBK01] et par transformation pignistique dans le cas crédibiliste [Sme05c].

Les travaux présentés dans ce manuscrit concernent un type de modèle particulier basé sur des *séquences d'états* évoluant temporellement. Dans de nombreuses applications, la notion de séquence est importante, par exemple pour la reconnaissance automatique de la parole [Rab89, NLP+02] et de l'écriture [VSD07], où les phrases sont définies comme des séquences de mots et de lettres, ou encore pour la reconnaissance de gestes complexes en utilisant une suite ordonnée de poses et de mouvements élémentaires [MHK06, WB01, NR02, WCK06, YPL07].

## 1.2 Méthodologies pour la reconnaissance de séquences

De nombreuses méthodes de reconnaissance de séquences ont été proposées pour modéliser des systèmes dont les états peuvent ou non être directement observables. Les états, tout comme les observations provenant des capteurs, peuvent être discrets ou continus et entâchés d'incertitude due aux conditions d'expérimentation.

L'évolution et la reconnaissance de séquences temporelles fait intervenir à la fois les observations courantes fournies par les capteurs mais aussi les connaissances sur les états aux instants précédents afin de rendre la reconnaissance plus robuste. Les liens temporels entre les états à plusieurs instants, les liens causaux entre les états ainsi que les liens entre les observations et les états sont représentés par divers formalismes, tels que la théorie des probabilités, des possibilités et des fonctions de croyance [KW99], capables de gérer l'incertitude sur les observations et sur les états.

Les méthodes de reconnaissance de séquences peuvent être groupées en deux catégories : les méthodes *symboliques*, proches des modèles et du raisonnement humain, et les méthodes *statistiques*, attachées aux données observées.

### 1.2.1 Méthodes symboliques

Dans la communauté de l'*Intelligence Artificielle*, les termes *chronique* et *scénario* sont largement utilisés pour décrire des séquences et un état de l'art sur les méthodes symboliques de reconnaissance de séquences est proposé dans la thèse de Vu [Vu04] et dans le manuscrit de l'Habilitation à Diriger des Recherches de Bremond [Bre07].

Les méthodes symboliques de reconnaissance de séquences reposent sur des langages formels permettant de retranscrire la connaissance des experts d'un domaine en un ensemble de règles. Ces

langages abstraits permettent de gérer des contraintes logiques et temporelles entre des événements et sont très proches des langages objets utilisés en programmation, comme en *Java* ou dans les automates de supervision. Lorsque les règles sont établies pour un problème particulier, un mécanisme de compilation génère un graphe minimal permettant de représenter graphiquement la séquence équivalente aux règles pour validation par l'expert. Les modèles de séquences sont donc construits à partir de connaissances expertes à un haut niveau sémantique et respecte un raisonnement humain.

Il existe de nombreux langages formels inspirés des travaux sur les intervalles temporels de Allen [All83, AF94, SW05]. Bobick et al. par exemple, proposent le langage *PNF* (*Past, Now, Future*) pour décrire des mouvements humains à l'aide d'un réseau de contraintes gérant les aspects temporels. Dousson et Ghallab [DGG93] ont développé un système de reconnaissance de chroniques (CRS : *Chronicle Recognition System*) basé sur le langage *IxTeT* et appliqué à des problèmes de supervision [CD00] notamment dans les réseaux de télécommunications [Dou96]. Le système CRS a été amélioré pour gérer les problèmes de hiérarchisation [DM07] ce qui est intéressant pour décrire des systèmes complexes.

Les méthodes basées sur la reconnaissance de chroniques peuvent mener à une explosion combinatoire car toutes les évolutions possibles sont stockées jusqu'à ce que l'une d'elles corresponde à une réalité : on parle de méthodes SPRS pour *Store Partially Recognized Scenarios*. Une solution à ce problème consiste en un système de comptage d'événements et de temporisation [Dou02] pour supprimer certaines évolutions. Une autre solution [Vu04, DM07] consiste à stocker les évolutions correspondant à une partie de chroniques déjà reconnues : on parle de méthodes STRS (*Store Totally Recognized Scenarios*). Un algorithme de type STRS inspiré des travaux de Dousson a été proposé par Vu, Bremond et Thonnat [VBT03, Vu04] pour la reconnaissance de scénarios dans les vidéos. Cet algorithme gère les problèmes de hiérarchisation pouvant simplifier la description et la reconnaissance de certains scénarios et autorise aussi des interactions entre plusieurs objets pouvant suivre des scénarios différents. D'autres langages associés à des ontologies de représentation ont été proposés dans le cadre de la surveillance [HTWM04] par Médioni, Hongeng et Nevatia [Hon02, HNB04, NHB04].

Le problème avec les méthodes symboliques concerne le fait que généralement elles ne gèrent pas l'incertitude : chaque état dans le graphe représente une variable binaire indiquant que l'événement associé à l'état apparaît ou n'apparaît pas. Ainsi, ces méthodes ignorent les mécanismes de bas niveau sémantique en charge de la détection des événements ce qui est discutable si l'on souhaite par exemple effectuer des adaptations, sous la forme de *feedback*, sur l'ensemble du système [Kun04]. De plus, les méthodes symboliques peuvent amener à des explosions combinatoires lors de la conception.

### 1.2.2 Méthodes statistiques

Dans les méthodes statistiques, des modèles de séquence sont estimés à partir de données d'apprentissage constituées généralement d'observations numériques pouvant être, ou non, étiquetées (cas d'un apprentissage supervisé vs. non-supervisé).

Un des challenges dans ces méthodes est de lier les observations numériques aux états tout en tenant compte de l'incertitude sur les observations et sur les états. Pour cela, on cherche généralement à estimer les distributions des observations conditionnellement aux états en utilisant, par exemple, l'algorithme EM (Espérance-Maximisation) et en supposant connue la forme des distributions (classiquement des Gaussiennes). La phase d'apprentissage permet par ailleurs d'estimer les liens statistiques entre les états à un même instant et à des instants consécutifs ce qui permet d'élaborer des *modèles d'évolution*. Ce type d'approche est par exemple très utilisé en reconnaissance de la parole [Rab89].

Lorsque la séquence a été modélisée (après la phase d'apprentissage), des outils permettent d'*inférer* les modèles de séquences à partir de nouvelles observations mesurées sur le système à étudier. Les critères d'inférence déterminent alors le meilleur modèle de séquence correspondant à ces observations.

Comme nous l'avons vu précédemment, le problème avec les méthodes statistiques concerne l'aspect boîtes noires car, bien que les modèles générés correspondent aux données d'apprentissage, ceux-ci ne sont généralement pas interprétables et leur mise à jour nécessite une nouvelle étape d'apprentissage dont la mise en œuvre est fastidieuse.

Les *modèles de Markov cachés* (HMM : *Hidden Markov Models*) [Rab89, BB02] représentent incontestablement l'outil statistique de reconnaissance de séquences le plus utilisé ces trois dernières décennies. Plus récemment, les modèles graphiques du type *réseaux Bayésiens dynamiques* [Mur02] sont apparus performants pour la modélisation, l'apprentissage et l'inférence. D'autres méthodes existent comme les réseaux de neurones artificiels [BB02] ou les automates à états finis [Mur95].

Les HMM sont adaptés aux cas où les états ne sont pas directement observables. Les HMM à observations discrètes ont été par exemple exploités en bio-informatique pour la reconnaissance de *séquences de base azotées* sur des brins d'ADN. Les observations discrètes possibles sont A, C, T et G et l'alternance de ces quatre bases azotées permet de décrire des séquences dans l'ADN, chaque séquence constituant un message porteur d'information génétique. Les HMM permettent d'*aligner* une séquence de bases azotées inconnue avec d'autres séquences connues afin de déterminer le rôle et les fonctions de la séquence analysée. Ce type de mécanisme permet de détecter de nouvelles molécules ou de décoder des génomes et a notamment été implanté sur des architectures à base de circuits électroniques reconfigurables [OSJM07] pour traiter de grandes quantités de données.

Les HMM, à observations discrètes ou continues, ont été beaucoup appliqués pour des problématiques liées à la reconnaissance automatique de la parole [Rab89, NLP+02] et de l'écriture [VSD07]. Dans ce contexte, une phrase est intuitivement représentée sous la forme d'une séquence de mots et de lettres. Les HMM ont été aussi employés[1] pour le traitement d'image [Pie03], l'indexation de vidéos [XMZY05], la détection et la classification d'ondes sismiques [BRS+07], la surveillance de patients cardiaques [ADB06] et de réseaux de télécommunications [VTW06].

Les méthodes statistiques offrent des outils adaptés pour la gestion de l'incertitude et qui sont généralement basés sur la théorie des probabilités. Cependant, les autres théories de l'incertain ont été peu évaluées sur le problème de la reconnaissance de séquences. Nous avons choisi la théorie des fonctions de croyance et en particulier le Modèle des Croyances Transférables (TBM : *Transferable Belief Model*) qui est un cadre de travail *générique* basé sur les *fonctions de croyance* et proposant des outils performants pour la gestion de l'incertitude et de l'imprécision. Des outils probabilistes puissants comme le Théorème de Bayes ont été étendus au TBM qui intègre par ailleurs des outils issus de la théorie des possibilités.

Le TBM permet de représenter explicitement le *doute* entre des états, ce qui s'avère intéressant pour la modélisation explicite des *transitions* entre états, que ce soit lors de l'apprentissage de modèles ou lors du processus de reconnaissance. De plus, le *conflit*, apparaissant entre les sources d'informa-

[1] Le lecteur intéressé par d'autres références et applications peut se rendre sur la page web intitulée « *10 years of HMM* » à l'adresse `http://www.tsi.enst.fr/~cappe/docs/hmmbib.html` contenant plus de 350 références sur l'utilisation des HMM de 1990 à 2000 ou bien se rendre sur le site web de l'institut IDIAP (`www.idiap.ch`) dont les activités de recherche portent sur ce type de méthodes appliquées à de nombreux domaines.

tion fusionnées pour détecter les états, est souligné apportant ainsi une information pertinente sur la *cohérence* du système de fusion.

## 1.3 Contribution

Dans cette thèse, nous proposons deux nouvelles méthodes de reconnaissance de séquences basées sur le TBM. La première est inspirée des méthodes symboliques et consiste en une machine à états finis déterministe. La seconde est une méthode statistique, crédibiliste et non-déterministe, qui consiste en la généralisation aux fonctions de croyance des processus sous-jacents aux modèles de Markov cachés initialement probabilistes.

Afin de valider nos approches, nous avons choisi d'appliquer ces méthodes sur la reconnaissance d'*actions* et d'*activités* humaines dans les *vidéos d'athlétisme*. Cette application entre dans le cadre de l'analyse de mouvements humains [MHK06] pour la gestion de base de vidéos sportives et des interfaces Homme-Machine [JS05]. Pour cela, nous avons mis en place un système permettant d'extraire des informations sur les mouvements des athlètes en collaboration[2] avec Dr. C. Panagiotakis et Pr. G. Tziritas de l'Université de Crète, de modéliser ces caractéristiques, et de reconnaître le type d'actions et d'activités effectuées par l'athlète à chaque image. Ce type de système pourrait par exemple être utilisé pour trier les événements sportifs sur un enregistrement d'émissions télévisées. Les méthodes de modélisation et de reconnaissance de séquences se veulent génériques et pourraient être appliquées à d'autres problèmes.

L'application est originale car les sauts d'athlétisme n'avaient jamais été utilisés jusqu'alors pour la reconnaissance de mouvements humains en partie à cause du fait que ces vidéos sont acquises en caméra mobile. Les phases de modélisation et de reconnaissance des mouvements et des séquences sont tout aussi originales puisqu'elles sont basées sur les fonctions de croyance et en particulier sur le Modèle des Croyances Transférables.

## 1.4 Organisation du mémoire

Le plan du manuscrit est le suivant :

**Chapitre 2 –** Le Modèle des Croyances Transférables est présenté. Nous décrivons tout d'abord les mécanismes de base puis des outils permettant de gérer des problèmes plus complexes de fusion d'information. Tout au long de ce chapitre, des exemples détaillés viennent illustrer le propos.

**Chapitre 3 –** Un algorithme est proposé pour filtrer des fonctions de croyance binaires évoluant au cours du temps. Celui-ci est appliqué à l'analyse de mouvements d'athlètes dans les vidéos.

**Chapitre 4 –** Une machine à états déterministe entrant dans la famille des méthodes symboliques est présentée pour reconnaître des séquences d'états. La connaissance sur les états est représentée par des fonctions de croyance temporelles. Cette méthode repose sur le filtre développé dans le chapitre 3 et les illustrations portent sur la reconnaissance de sauts d'athlétisme dans les vidéos.

**Chapitre 5 –** Les mécanismes sous-jacents aux modèles de Markov cachés probabilistes sont en partie généralisés aux fonctions de croyance. Les modèles de Markov cachés crédibilistes, faisant

[2] La collaboration a été effectuée dans le cadre du Réseau d'Excellence Européen SIMILAR.

partie des méthodes statistiques, permettent de reconnaître des séquences d'états de façon stochastique où la connaissance sur les états est représentée par des fonctions de croyance. Des illustrations portant sur l'analyse de mouvements d'athlètes dans les vidéos sont présentées.

**Chapitre 6 –** Les méthodes développées dans les chapitres 3 à 5 sont évaluées sur une base de $74$ vidéos d'athlétisme et comparées aux modèles de Markov cachés probabilistes.

# Chapitre 2

# Le Modèle des Croyances Transférables pour la représentation des connaissances, la combinaison et la prise de décision

Le *Modèle des Croyances Transférables* (TBM : *Transferable Belief Model*) est un cadre formel générique développé par Ph. Smets [SK94] pour la représentation et la combinaison des connaissances. Le TBM est basé sur les fonctions de croyance et propose un ensemble d'opérateurs permettant de combiner des fonctions de croyance fournies par des sources d'information pouvant être complémentaires, redondantes et éventuellement non-indépendantes.

Les mécanismes de raisonnement du TBM peuvent être regroupés en deux niveaux, représentant une abstraction des mécanismes et illustrés sur la figure 2.1 :

- Le niveau *crédal* (du latin « credo » signifiant « je crois ») siège de la représentation des connaissances (partie statique), des combinaisons et du raisonnement sur ces connaissances (partie dynamique),
- Le niveau *pignistique* (du latin « pignus » signifiant « un pari ») et impliquant la prise de décision en prenant éventuellement en compte le risque d'effectuer certaines actions suite aux décisions.

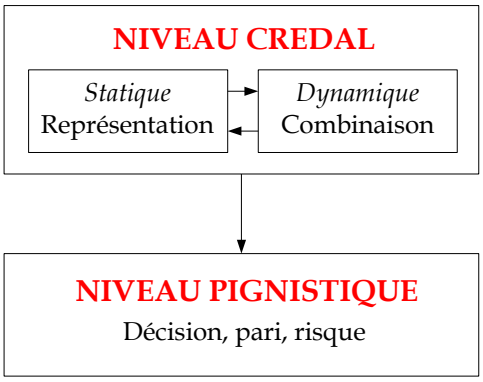



**FIG. 2.1 :** *Représentation abstraite des mécanismes dans le Modèle des Croyances Transférables.*

Cette décomposition en deux niveaux différencie le TBM des autres théories et méthodes, en particulier la théorie des probabilités au sein de laquelle la représentation des connaissances, la

combinaison et la décision sont effectuées sans distinguer les niveaux d'abstraction.

Ce chapitre est divisé en deux sections : la première section est consacrée à la description des outils de base du TBM et la seconde section est dédiée à des opérateurs permettant de régler des problèmes de fusion complexes :

| **Section 2.1 : Outils de base** | **Section 2.2 : Outils avancés** |
|---|---|
| Représentation de la connaissance avec les fonctions de croyance | Conditionnement et Théorème de Bayes Généralisé |
| Combinaisons : conjonction, disjonction, fiabilité simple | Changement de cadre de discernement |
| Le principe du minimum d'information et son utilisation | Combinaisons dans le cas de sources non-distinctes |
| Prise de décision | Fiabilité contextuelle |
| Gestion du conflit | |

## 2.1 Outils de base

Cette section répond aux questions suivantes (Fig. 2.2) :
- Comment *représenter* la connaissance d'une source d'information sous forme de fonctions de croyance et quels avantages peut-on tirer d'une telle représentation ?
- Comment *combiner* plusieurs sources de fonctions de croyance afin de résumer l'information et d'améliorer la prise de décision ?
- Comment *prendre une décision* à partir de fonctions de croyance ?

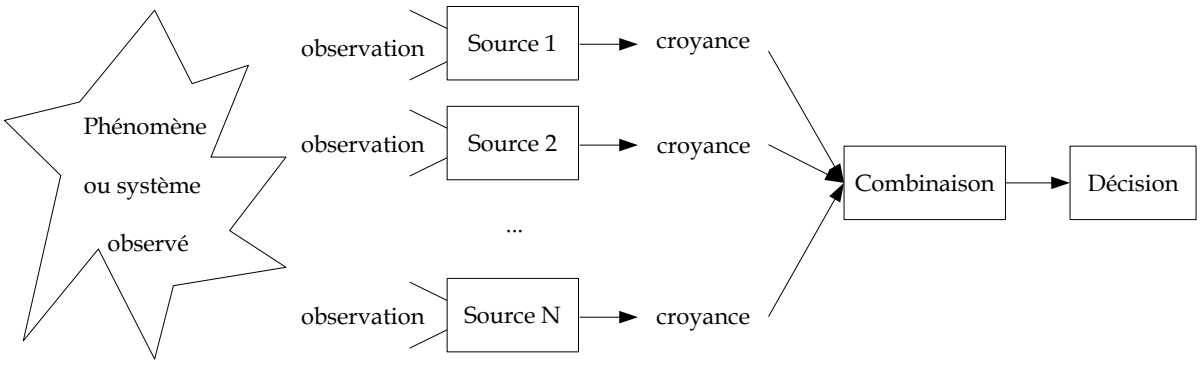



**FIG. 2.2 :** *Fusion d'information : des sources de croyance émettent une opinion pondérée sur l'état du système observé. Les différentes sources sont combinées afin de synthétiser la compréhension du système et améliorer la décision sur l'état du système après combinaison.*

Tout d'abord, nous présentons les *concepts* importants à partir d'un exemple simple : le lancé de dé. Puis, nous décrivons le *formalisme mathématique* des fonctions de croyance ainsi que les *outils de base* permettant de *combiner* ces fonctions et de prendre des décisions en rapport avec leurs valeurs.

### 2.1.1 Présentation des concepts sur un cas d'école : le lancé de dé

Nous considérons un dé à six faces (numérotées $1, 2 \ldots 6$) lancé par un joueur. Les chiffres possibles pouvant être tirés par le dé sont appelés *hypothèses* et l'ensemble des hypothèses forment le *cadre de discernement* généralement noté $\Omega$, par exemple $\Omega = \{F_1, F_2, F_3, F_4, F_5, F_6\}$ où $F_i$ est la face montrant le chiffre $i$. Les hypothèses sont supposées *exclusives* signifiant que deux hypothèses ne peuvent être vraies simultanément. L'exclusivité se traduit par une intersection nulle entre les hypothèses, par exemple $\{F_1\} \cap \{F_3\} = \emptyset$.

La face tirée à l'aide du dé est en partie cachée et seul le point central est visible. Le joueur émet alors une opinion pondérée sur le chiffre du dé ce qui fait de lui une *source* d'information. Au vu de la connaissance disponible après le lancé du dé, le joueur affirme que le chiffre est soit $F_1$, soit $F_3$ ou soit $F_5$. Comment modéliser cette connaissance ?

Dans le cadre de la *théorie des probabilités*, la réponse est donnée par le principe d'*équiprobabilité* : chaque face se voit affectée une même probabilité de $\frac{1}{3}$ telle que la somme des probabilités soit égale à 1. La *distribution* de probabilités pour ce jet de dé est alors égale à la distribution uniforme :

$$P^{\Omega}(F_1) = P^{\Omega}(F_2) = P^{\Omega}(F_3) = \frac{1}{3}$$

La probabilité est affectée seulement à des hypothèses singletons.

Dans le cadre de la *théorie des fonctions de croyance*, la connaissance est modélisée par une *distribution de masses de croyance*. Dans l'exemple du dé, la totalité de la masse de croyance, c'est à dire une unité comme en probabilité, est affectée à la *proposition* $\{F_1, F_3, F_5\}$ formée d'une *union d'hypothèses* et la distribution se réduit alors à un élément :

$$m^{\Omega}(\{F_1, F_3, F_5\}) = 1$$

La masse affectée à la proposition $\{F_1, F_3, F_5\}$ *ne donne aucune information* concernant la croyance sur chacune des hypothèses $F_1$, $F_3$ ou $F_5$ composant cette proposition. On dit qu'une masse de croyance modélise *explicitement le doute* (ou l'ignorance) sur cet ensemble d'hypothèses. En comparaison, dans le cadre de la théorie des probabilités, la valeur de la probabilité sur une union d'hypothèses est déduite *implicitement* de la probabilité sur les hypothèses singletons par une somme des probabilités à laquelle on retire la partie commune, c'est à dire la probabilité de l'intersection ou encore du couple. On dit qu'une mesure de probabilité est *additive*. Par exemple :

$$P^{\Omega}(\{F_3, F_5\}) = P^{\Omega}(F_3) + P^{\Omega}(F_5) - P^{\Omega}(\{(F_3, F_5)\})$$

où la probabilité sur l'intersection ($P^{\Omega}(\{(F_3, F_5)\})$) représente la partie commune des événements « la face est $F_3$ » et « la face est $F_5$ » qu'il est nécessaire de retirer pour éviter de compter les événements deux fois. Si les hypothèses sont exclusives, comme c'est le cas tout au long de ce manuscrit, alors :

$$\{F_3\} \cap \{F_5\} = \emptyset \;\Rightarrow P^{\Omega}(\{(F_3, F_5)\}) = 0 \;\Rightarrow P^{\Omega}(\{F_3, F_5\}) = P^{\Omega}(F_3) + P^{\Omega}(F_5)$$

car $P^{\Omega}(\emptyset) = 0$ par définition de la mesure de probabilité.

Une fonction de masses *n'est pas additive* ce qui signifie que la masse sur une union d'hypothèses n'est pas égale à la somme des masses des hypothèses composant l'union [KW99] et cette propriété permet de représenter explicitement le doute entre des hypothèses.

Outre la représentation du doute, un des avantages des fonctions de croyance est leur généricité. En effet, elles représentent mathématiquement un sur-ensemble des fonctions de probabilités : une distribution de masses où seules les hypothèses singletons ont une masse non nulle peut être interprétée comme une distribution de probabilités. Nous verrons aussi que les fonctions de croyance généralisent mathématiquement les fonctions de possibilités [KW99]. Chacune de ces fonctions (probabilités, possibilités et croyances) possède des caractéristiques [KW99] et permettent de modéliser la connaissance différemment.

La section suivante est dédiée à la présentation du formalisme mathématique des fonctions de croyance dans le cadre du Modèle des Croyances Transférables (TBM). Notons que plusieurs modèles d'interprétation des fonctions de croyance ont été proposés : le Modèle de Dempster [Dem67, Dem68], le Modèle de Shafer [Sha76], le Modèle des Croyances Transférables [SK94] et le Modèle des Hints [KM95] entre autres, chacun disposant de spécificités propres comme des opérateurs de combinaison ou de décision. Nous nous plaçons dans le cadre axiomatisé et formalisé du Modèle des Croyances Transférables développé par Ph. Smets.

### 2.1.2 Formalisme des fonctions de croyance dans le TBM

La figure 2.2 représente un synopsis de fusion d'information. L'objectif du système est de déterminer l'état *réel* $\omega_0$ d'un système étudié à partir d'observations.

L'état du système est supposé prendre des valeurs *discrètes* appelées *hypothèses* $\omega_k$ et l'ensemble des $K$ hypothèses possibles pouvant expliquer l'état du système est appelé *cadre de discernement* (ou *univers de discours*) généralement noté :

$$\Omega = \{\omega_1 \dots \omega_k \dots \omega_K\} = \bigcup_{k=1}^{K} \{\omega_k\}$$

Le nombre d'hypothèses $|\Omega| = K$ composant $\Omega$ est appelé *cardinal* et les hypothèses sont considérées *exclusives*.

L'observation du système est assurée par des *agents*[1] qui fournissent, entre autres choses, une opinion pondérée représentée par une fonction de croyance (Fig. 2.2) concernant l'état réel $\omega_0 \subseteq \Omega$ du système. Les agents jouent le rôle de *sources* de croyance.

#### 2.1.2.1 Définition de la fonction de masse

Les fonctions de croyance peuvent être représentées de différentes manières. La représentation la plus communément utilisée est celle de la *fonction de masses de croyance* notée $m^\Omega$ définie par :

$$\begin{array}{rcccl} m^\Omega & : & 2^\Omega & \rightarrow & [0,1] \\ & & B & \mapsto & m^\Omega(B) \end{array} \tag{2.1}$$

où l'espace puissance $2^\Omega = \{\emptyset, \{\omega_1\}, \{\omega_2\}, \{\omega_1, \omega_2\}, \{\omega_3\}, \{\omega_1, \omega_3\}, \{\omega_2, \omega_3\}, \{\omega_1, \omega_2, \omega_3\}, \dots, \{\omega_1, \dots, \omega_K\}\}$ rassemble tous les sous-ensembles possibles formées des hypothèses et unions d'hypothèses de $\Omega$. Notons que la source de croyance $S$ est souvent inscrite en indice de la fonction de masses, par exemple $m_S^\Omega$.

[1] Terme générique issu de la communauté de l'*Intelligence Artificielle* représentant des capteurs, des humains ...

Une proposition $B \subseteq \Omega$, par exemple $B = \{\omega_1, \omega_3\}$, représente explicitement le doute entre les hypothèses composant $B$ et la masse de croyance $m^\Omega(B)$ allouée à $B$ ne donne aucune information sur les hypothèses et sous-ensembles composant $B$ [KW99]. Cette caractéristique provient de la nature non-additive des fonctions de masses et représente, comme nous l'avons vu dans l'exemple du *lancé de dé*, une différence fondamentale avec la théorie des probabilités où la mesure utilisée est additive.

Une *distribution de masses* de croyance (BBA : *basic belief assignment*) fournie par une source $S$ est un ensemble de masses de croyance concernant des propositions $B \subseteq \Omega$ (ou de manière équivalente $B \in 2^\Omega$) vérifiant :

$$\sum_{B \subseteq \Omega} m_S^\Omega(B) = 1 \tag{2.2}$$

La masse $m_S^\Omega(B)$ représente la part de croyance de la source $S$ en la proposition « $\omega_0 \in B$ » où $\omega_0$ est l'état réel du système observé. Un élément $B$ dont la masse est non nulle ($m_S^\Omega(B) > 0$) est appelé *élément focal* de la BBA. Un exemple de BBA est donné colonne « quelconque » (colonne 6) du tableau 2.1.

Quelques exemples particuliers de BBA sont à noter :

**BBA normale :** la masse sur l'ensemble vide est nulle, i.e. $m^\Omega(\emptyset) = 0$. Dans le cas contraire, la BBA est dite *sous-normale*. Le passage d'une BBA *sous-normale* $m^\Omega$ à une BBA *normale* $m_*^\Omega$ peut être effectué par le processus de *normalisation* défini par :

$$\begin{aligned} m_*^\Omega(B) &= \frac{m^\Omega(B)}{1 - m^\Omega(\emptyset)}, \quad \forall B \subseteq \Omega, B \neq \emptyset \\ m_*^\Omega(\emptyset) &= 0 \end{aligned} \tag{2.3}$$

**BBA catégorique :** un seul élément de l'ensemble des propositions est un élément focal, i.e. $m^\Omega(B) = 1, B \subseteq \Omega$ et $m^\Omega(C) = 0, C \subseteq \Omega$ et $C \neq B$.

**BBA vide :** l'unique élément focal est $\Omega$, i.e. $m^\Omega(\Omega) = 1$ et indique que la source n'a aucune information sur l'état $\omega_0$ (dernière colonne intitulée « vide » du tableau 2.1).

**BBA dogmatique :** la proposition $\Omega$ n'est pas un élément focal, i.e. $m^\Omega(\Omega) = 0$ (colonne « bayésienne » du tableau 2.1).

**BBA simple :** il existe au plus deux éléments focaux dont $\Omega$ (la BBA est donc non-dogmatique), i.e. pour $\eta \in [0, 1], m^\Omega(B) = 1 - \eta$ pour $B \subset \Omega$ et $m^\Omega(\Omega) = \eta$. Ces BBA sont généralement notées $B^\eta$.

**BBA Bayésienne :** les éléments focaux sont des singletons et donc la BBA $m^\Omega$ est une distribution de probabilités avec $m^\Omega(\{\omega_k\}) > 0, \omega_k \in \Omega$. Un exemple d'une telle BBA est donné dans la $4^e$ colonne intitulée « Bayésienne » du tableau 2.1. Dans la littérature, des BBA *quasi-Bayésiennes* sont fréquemment utilisées dont les éléments focaux sont les singletons ainsi que $\Omega$.

**BBA consonante :** les éléments focaux sont emboités, i.e. $B \subseteq C \subseteq D \ldots$ Un exemple de BBA consonante est donné dans la $5^e$ colonne intitulée « consonante » du tableau 2.1. Ces BBA ont des caractéristiques communes avec les possibilités [Sme00b, Sme00c, DGPS01].

**BBA sans conflit interne :** elles sont caractérisées par des éléments focaux dont les intersections ne sont pas vides, i.e. $\cap\{\omega \subseteq \Omega / m^\Omega(\omega) > 0\} \neq \emptyset$. Par conséquent ces BBA sont normales et les BBA consonantes sont sans conflit interne.

Lorsque le cadre de discernement est *exhaustif*, cela signifie qu'il contient forcément l'hypothèse permettant de décrire le système observé (sinon le cadre de discernement n'est pas adapté au problème traité). Dans ce cas, la masse sur l'ensemble vide est nulle, i.e. $m^{\Omega}(\emptyset) = 0$, et la masse est *normale*. On parle alors de *monde fermé* (le « monde » étant le cadre de discernement), comme c'est le cas dans le modèle de Shafer [Sha76]. Dans le cadre du TBM, le cadre de discernement peut être *non exhaustif* et donc la masse sur l'ensemble vide peut être non nulle (et la masse *sous-normale*). On parle alors de *monde ouvert*. Le monde ouvert permet par exemple de redéfinir le cadre de discernement en ajoutant une hypothèse [DS04] en fonction de la valeur de la masse sur l'ensemble vide. Cette nouvelle hypothèse peut correspondre à une *hypothèse de rejet*. Une masse positive sur l'ensemble vide peut être interprétée de différentes manières et peut poser quelques problèmes comme nous le verrons dans le paragraphe 2.1.6. Le passage du monde ouvert au monde fermé consiste à *redistribuer* la masse de l'ensemble vide sur les autres sous-ensembles de l'espace puissance $2^{\Omega}$. Plusieurs *règles de redistribution* existent [LCV02] dont la *normalisation* présentée précédemment.

| $B \subseteq \Omega$ | **Précise** | **Imprécise** | **Bayésienne** | **Consonante** | **Quelconque** | **Vide** |
|---|---|---|---|---|---|---|
| $\emptyset$ | 0 | 0 | 0 | 0 | 0.1 | 0 |
| $\{\omega_1\}$ | 0 | 0 | 0.3 | 0.35 | 0.07 | 0 |
| $\{\omega_2\}$ | 0 | 0 | 0.6 | 0 | 0.12 | 0 |
| $\{\omega_1, \omega_2\}$ | 0 | 0 | 0 | 0.2 | 0.22 | 0 |
| $\{\omega_3\}$ | 1 | 0 | 0.1 | 0 | 0.31 | 0 |
| $\{\omega_1, \omega_3\}$ | 0 | 1 | 0 | 0 | 0.08 | 0 |
| $\{\omega_2, \omega_3\}$ | 0 | 0 | 0 | 0 | 0.01 | 0 |
| $\{\omega_1, \omega_2, \omega_3\}$ | 0 | 0 | 0 | 0.45 | 0.09 | 1 |

**TAB. 2.1 :** *Exemple de BBA $m^{\Omega}$ exprimée sur un cadre de discernement $\Omega = \{\omega_1, \omega_2, \omega_3\}$. Le cas* crédibiliste *(fonction de masses de croyance quelconque) permet de couvrir tous les cas ($6^e$ colonne), i.e. de la connaissance précise et certaine (colonne 2) à l'ignorance totale (colonne 7). Inspiré de [VS05].*

#### 2.1.2.2 Transformations de la fonction de masses

A partir de la fonction de masses $m^{\Omega}$, d'autres fonctions en *relation biunivoque* peuvent être déduites : elles représentent la *même information* mais sous une forme différente. Ces fonctions ont diverses interprétations mais en ce qui nous concerne, nous les utiliserons essentiellement pour simplifier les calculs de combinaisons. Elles sont toutes définies de $2^{\Omega}$ vers $[0, 1]$ mais la somme des éléments focaux n'est pas forcément unitaire. Les fonctions élémentaires que nous utiliserons sont les suivantes :

**Plausibilité** $pl$ **:** la plausibilité d'un élément $B$ est la part maximale de croyance qui pourrait soutenir $B$ :

$$pl^{\Omega}(B) = \sum_{C \cap B \neq \emptyset} m^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.4a}$$

$$m^{\Omega}(B) = \sum_{C \subseteq B} (-1)^{|B|-|C|+1} pl^{\Omega}(\overline{C}), \quad \forall B \subseteq \Omega \tag{2.4b}$$

**Crédibilité (ou *croyance*)** $bel$ **:** la crédibilité d'un élément $B$ est la part de croyance spécifiquement (sans le conflit) allouée à $B$ :

$$bel^{\Omega}(B) = \sum_{\emptyset \neq C \subseteq B} m^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.5a}$$

$$m^{\Omega}(B) = \sum_{C \subseteq B} (-1)^{|B|-|C|} bel^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.5b}$$

**Implicabilité** $b$ **:** l'implicabilité d'un élément $B$ est la somme des masses allouées aux sous-ensembles de $B$ tels que leur véracité implique la véracité de $B$ :

$$b^{\Omega}(B) = \sum_{C \subseteq B} m^{\Omega}(C) = bel^{\Omega}(B) + m^{\Omega}(\emptyset), \quad \forall B \subseteq \Omega \tag{2.6a}$$

$$m^{\Omega}(B) = \sum_{C \subseteq B} (-1)^{|B|-|C|} b^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.6b}$$

**Communalité** $q$ **:** la communalité d'un élément $B$ représente la somme des masses allouées aux sur-ensembles de $B$ et donc qui ont $B$ en commun :

$$q^{\Omega}(B) = \sum_{C \supseteq B} m^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.7a}$$

$$m^{\Omega}(B) = \sum_{B \subseteq C} (-1)^{|C|-|B|} q^{\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.7b}$$

Le tableau 2.2 donne une indication sur l'utilisation de ces fonctions dans le cadre de cette thèse. Un résumé de ces transformées est donné dans [Sme02a] et l'outil graphique TBMlab[2] contient quelques procédures Matlab© pour les conversions.

| Sigle | Nom | Utilité |
|---|---|---|
| $m$ | Masse | Représentation de la connaissance |
| $pl$ | Plausibilité (Eq. 2.4a) | Théorème de Bayes Généralisé, vraisemblances |
| $bel$ | Crédibilité (Eq. 2.5a) | Représentation de la connaissance |
| $b$ | Implicabilité (Eq. 2.6a) | Règle de combinaison disjonctive (DRC) |
| $q$ | Communalité (Eq. 2.7a) | Règle de combinaison conjonctive (CRC) |

**TAB. 2.2 :** *Utilisation des fonctions de croyance dans le cadre de nos travaux.*

**Exemple 1.** *Le tableau 2.3 présente le résultat de la transformation de la BBA donnée de la colonne « quelconque » du tableau 2.1 en plausibilité ($pl$), crédibilité ($bel$), implicabilité ($b$) et communalité ($q$). Il est important de noter que le calcul de la crédibilité $bel$ nécessite la normalisation préalable (Eq. 2.3) de la distribution de masses pour éliminer la masse sur l'ensemble vide.*

### 2.1.3 Le principe du minimum d'information

*Le principe du minimum d'information* (PMI) est au cœur d'un grand nombre de mécanismes développés dans le TBM. L'idée est la suivante : lorsque l'on doit choisir une BBA parmi un ensemble

[2]Disponible à l'adresse `http://iridia/ulb.ac.be/~psmets/`.

| | $m$ | $pl$ | $bel$ | $b$ | $q$ |
|---|---|---|---|---|---|
| $\emptyset$ | 0.1 | 0 | 0 | 0.1 | 1 |
| $\{\omega_1\}$ | 0.07 | 0.46 | 0.08 | 0.17 | 0.46 |
| $\{\omega_2\}$ | 0.12 | 0.44 | 0.13 | 0.22 | 0.44 |
| $\{\omega_1, \omega_2\}$ | 0.22 | 0.59 | 0.45 | 0.51 | 0.31 |
| $\{\omega_3\}$ | 0.31 | 0.49 | 0.34 | 0.41 | 0.49 |
| $\{\omega_1, \omega_3\}$ | 0.08 | 0.78 | 0.51 | 0.56 | 0.17 |
| $\{\omega_2, \omega_3\}$ | 0.01 | 0.83 | 0.48 | 0.54 | 0.1 |
| $\{\omega_1, \omega_2, \omega_3\}$ | 0.09 | 0.9 | 1 | 1 | 0.09 |

**TAB. 2.3 :** *Résultat de la transformation de la BBA donnée dans la* $6^e$ *colonne du tableau 2.1 en plausibilité (pl), crédibilité (bel), implicabilité (b) et communalité (q).*

de BBA possibles, le PMI impose de choisir celle qui est la *moins engagée* ou, autrement dit, la *moins informative*.

Une approche *quantitative* consiste à calculer le *taux d'incertitude* des BBA et de choisir celle qui maximise l'incertitude minimisant ainsi le parti pris. Un grand nombre de méthodes ont été proposées pour mesurer le taux d'incertitude à partir d'informations imprécises [KW99, Har99] (voir aussi la mesure proposée dans [DM04] dans le cadre du *clustering*). La plus répandue des mesures est la *non-spécificité* [DP86b] basée sur la cardinalité des éléments focaux pondérée par la valeur des masses :

$$N(m) = \sum_{\emptyset \neq A \subseteq \Omega} m(A) \cdot \log_2(|A|) \tag{2.8}$$

Cette mesure est bornée sur $[0, \log_2(|\Omega|)]$.

Plus la mesure de non-spécificité est grande moins la BBA est informative. Par exemple, la BBA vide est totalement non-spécifique tandis qu'une BBA précise (telle qu'une BBA Bayésienne) est totalement spécifique. Le PMI basé sur la non-spécificité maximise donc la valeur de ce critère jouant un rôle similaire au maximum d'entropie en théorie de l'information. Il est par ailleurs possible d'utiliser des relations d'ordre entre des fonctions de croyance (approche *qualitative* du PMI) [DP86a, Yag87, Sme00c, Sme00b].

**Exemple 2.** *Les BBA données dans les colonnes intitulées « Bayésienne », « consonante » et « quelconque » (*$4^e, 5^e$ *et* $6^e$ *colonnes) du tableau 2.1 ont pour non-spécificité* $N(m_{4^e}) = 0$*,* $N(m_{5^e}) \approx 0.91$ *et* $N(m_{6^e}) \approx 0.45$ *respectivement. La BBA consonante est ici la moins informative. La non-spécificité d'une BBA Bayésienne est nulle car les éléments focaux sont des singletons ce qui traduit un parti pris total.*

### 2.1.4 Règles de combinaison de base

La figure 2.2 illustre un cas de *fusion d'information* qui *consiste à combiner des informations hétérogènes issues de plusieurs sources afin d'améliorer la prise de décision* [Blo05]. Plusieurs opérateurs de fusion de BBA ont été développés dans le cadre du TBM. Les régles de base présentées dans ce paragraphe sont basées sur l'hypothèse que les sources de croyance sont *distinctes* [Sme93, YSM01, YSM02a, YSM02b], concept proche de l'*indépendance* en statistique.

#### 2.1.4.1 La CRC pour combiner des sources distinctes et fiables

Soient deux BBA $m_1^\Omega$ et $m_2^\Omega$ distinctes et définies sur le même cadre de discernement $\Omega$. Les sources sont supposées *fiables* [Sha76]. La *règle de combinaison conjonctive* (CRC : *conjunctive rule of combination*), notée $\textcircled{\cap}$, des BBA $m_1^\Omega$ et $m_2^\Omega$, ou de leur communalité associée $q_1^\Omega$ et $q_2^\Omega$, est définie $\forall B, C, D \subseteq \Omega$ par les équations suivantes :

$$\begin{array}{rclcl} m_{1\textcircled{\cap}2}^\Omega(D) & = & (m_1^\Omega \textcircled{\cap} m_2^\Omega)(D) & = & \displaystyle\sum_{B\cap C=D} m_1^\Omega(B)\cdot m_2^\Omega(C) \\ q_{1\textcircled{\cap}2}^\Omega(D) & = & (q_1^\Omega \textcircled{\cap} q_2^\Omega)(D) & = & q_1^\Omega(D)\cdot q_2^\Omega(D) \end{array} \tag{2.9}$$

L'intersection intervenant dans la CRC joue un effet de *spécialisation* (voir les références et définitions de [Sme02a]) en transférant la masse sur des sous-ensembles de cardinalité plus faible. Ainsi, lorsque l'une des BBA est Bayésienne alors le résultat est une BBA Bayésienne. Notons que l'utilisation des communalités réduit la CRC à un produit membre à membre simplifiant ainsi les calculs.

Les propriétés notables de la règle CRC sont les suivantes :

**La CRC n'est pas idempotente :** la CRC est associative et commutative mais elle n'est pas idempotente ce qui signifie que $(m_1^\Omega \textcircled{\cap} m_1^\Omega) \neq m_1^\Omega$ et par conséquent il n'est pas possible de combiner des sources dépendantes ou en interaction. Dans la section 2, nous présenterons un opérateur récent proposé par Denœux [Den07] qui est associatif, commutatif et idempotent. Notons que l'élément neutre est la BBA vide et que $m^\Omega(\emptyset) = 1$ est l'élément absorbant.

**La CRC souligne le conflit :** lorsque l'intersection des éléments $B$ et $C$ est vide ($B \cap C = \emptyset$), les sources sont conflituelles. Il est nécessaire d'identifier les raisons de ce conflit car l'ensemble vide est absorbant par l'intersection. Des règles de redistribution ont été proposées [LCV02] pour annuler son effet mais il est généralement préférable d'analyser ses origines (sujet de § 2.1.6).

**La BBA issue d'une CRC peut être sous-normale :** la masse sur l'ensemble vide peut être non nulle [Sme92] si les sources sont conflictuelles. Si la normalisation est effectuée alors la CRC revient à la *règle orthogonale* de Dempster [Dem67, Dem68] (notée $\oplus$) utilisée dans le modèle de Shafer [Sha76, Sme94a]. La CRC normalisée est définie par :

$$m_{1\oplus 2}^\Omega(D) = (m_1^\Omega \oplus m_2^\Omega)(D) = \begin{cases} \dfrac{m_{1\textcircled{\cap}2}^\Omega(D)}{1 - m_{1\textcircled{\cap}2}^\Omega(\emptyset)}, & \forall D \subseteq \Omega \quad \text{si } D \neq \emptyset \\ 0 & \text{si } D = \emptyset \end{cases} \tag{2.10}$$

**La CRC est une opération linéaire :** elle peut donc être décrite sous forme matricielle [Sme02a]. Cette notation permet de mettre en évidence l'idée de *transfert des masses* sous-jacent au Modèle des Croyances *Transférables* (voir [Mer06] pour une application concernant la reconnaissance d'adresses postales).

#### 2.1.4.2 Le conditionnement pour combiner des informations certaines

La CRC permet de définir l'opération de *conditionnement*. Conditionner une BBA $m^\Omega$ par un élément $B \subseteq \Omega$ consiste à restreindre le cadre des propositions possibles $2^\Omega$ à celles ayant une intersection non nulle avec $B$. Ainsi, lors d'un conditionnement, les masses affectées à $C \subseteq \Omega$ sont transférées sur $C \cap B$. La *BBA conditionnelle* qui en résulte est notée $m^\Omega[B]$ (les crochets traduisent

le conditionnement) et ces éléments sont donnés par la règle de *conditionnement de Dempster non normalisée* :

$$m^{\Omega}[B](C) = \sum_{D \subseteq \overline{B}} m^{\Omega}(C \cup D), \quad \forall C \subseteq B \tag{2.11}$$

L'opération de conditionnement est particulièrement adaptée dans les cas où l'on dispose d'une information *certaine* (mais qui peut être imprécise, par exemple la colonne 3 du tableau 2.1). Le conditionnement revient à effectuer une combinaison conjonctive avec une *BBA catégorique sur* $B$ (i.e. $m^{\Omega}(B) = 1$) et donc la BBA issue d'un conditionnement est souvent sous-normale (la gestion du conflit est décrite dans § 2.1.6).

#### 2.1.4.3 La DRC pour combiner des sources distinctes et non-fiables

La *règle de combinaison disjonctive* (DRC : *disjunctive rule of combination*), notée $\textcircled{\cup}$, a été proposée [DP86b, Sme93] pour combiner des BBA dont l'une au moins est fiable, sans pouvoir quantifier la fiabilité ni savoir quelle source est fiable. Elle est définie $\forall D \subseteq \Omega$ par les équations suivantes :

$$\begin{array}{rclcl} m^{\Omega}_{1 \textcircled{\cup} 2}(D) & = & (m^{\Omega}_1 \textcircled{\cup} m^{\Omega}_2)(D) & = & \displaystyle\sum_{B \cup C = D} m^{\Omega}_1(B) \cdot m^{\Omega}_2(C) \\ b^{\Omega}_{1 \textcircled{\cup} 2}(D) & = & (b^{\Omega}_1 \textcircled{\cup} b^{\Omega}_2)(D) & = & b^{\Omega}_1(D) \cdot b^{\Omega}_2(D) \end{array} \tag{2.12}$$

La DRC est une règle associative, commutative mais non idempotente (nous décrirons un opérateur disjonctif idempotent dans la section 2) dont l'élément neutre est la BBA $m^{\Omega}(\emptyset) = 1$ et l'élément absorbant la BBA vide. Cette règle est conservatrice car elle transfère les masses sur des sur-ensembles des éléments focaux. Par conséquent, les éléments focaux résultants peuvent être de cardinal élevé. On dit que la BBA résultante est moins spécialisée que les BBA d'origine ou encore que la DRC est un processus de *généralisation* [Sme02a] (le contraire de la spécialisation). Notons que l'utilisation des implicabilités ($b$) permet de réduire la DRC à un produit membre à membre et que la DRC est une opération linéaire au même titre que la CRC pouvant être décrite sous forme matricielle [Sme02a].

**Remarque 2.1.** *La dualité des règles* $\textcircled{\cap}$ *et* $\textcircled{\cup}$ *est mise en valeur dans les règles de* De Morgan *s'appliquant aux fonctions de croyance [DP86b].*

#### 2.1.4.4 Règle d'affaiblissement *simple* pour intégrer la fiabilité des sources

L'application de la DRC lorsque les sources de BBA ne sont pas fiables peut s'avérer trop conservatrice pouvant mener à une BBA complètement non-informative (BBA vide). Dans certaines applications, il est cependant possible de *quantifier la fiabilité* ce qui permet d'appliquer une règle de combinaison moins conservatrice dans le cas de sources non fiables.

La prise en compte de la fiabilité dans le cadre des fonctions de croyance porte le nom d'*affaiblissement* car le processus consiste à pondérer la masse des éléments d'une BBA. Les premiers travaux sur l'affaiblissement dans le cadre des fonctions de croyance ont été développés par Shafer [Sha76], axiomatisés par Smets [Sme93] et généralisés par Mercier et Denœux [MQD07, Mer06]. L'affaiblissement généralisé a été appelé *affaiblissement contextuel*.

L'affaiblissement « simple » (de Shafer) d'une BBA $m^\Omega$ est défini comme la pondération de chaque masse $m^\Omega(B)$ de la distribution par un coefficient $(1-\alpha)$ appelé *fiabilité* et où $\alpha \in [0,1]$ est le *taux d'affaiblissement*. Formellement :

$$\begin{array}{rcl} {}^{\alpha}m^\Omega(B) & = & (1-\alpha)\cdot m^\Omega(B), \quad \forall B \subset \Omega \\ {}^{\alpha}m^\Omega(\Omega) & = & (1-\alpha)\cdot m^\Omega(\Omega) + \alpha \end{array} \tag{2.13}$$

Par le PMI, le reste de la masse (après pondération) est transféré sur l'élément d'ignorance $\Omega$. Plus la source est fiable (et donc plus le taux d'affaiblissement $\alpha$ est faible), moins la BBA est modifiée alors que, plus la fiabilité diminue, plus la BBA issue de l'affaiblissement tend vers la BBA vide.

### 2.1.5 Prise de décision à partir de fonctions de croyance

Les systèmes de fusion d'information, qu'ils soient basés sur la théorie des probabilités, des possibilités ou des fonctions de croyance, ont pour finalité la prise de décision. Prendre une décision consiste à choisir une hypothèse sur un *cadre de pari*, généralement le cadre de discernement $\Omega$. La prise de décision peut être réalisée de façon automatique ou laissée à la responsabilité de l'utilisateur final (par exemple dans le cas de *l'aide au diagnostic* dans le domaine de la Médecine).

#### 2.1.5.1 Prise de décision à partir des probabilités pignistiques

Dans le cadre du TBM, la phase de décision s'appuie sur la *distribution de probabilités pignistiques* [Sme05c] notée $\mathbf{BetP}\{m^\Omega\}$ obtenue à partir de la BBA $m^\Omega$. La transformée pignistique consiste à répartir de manière équiprobable la masse d'une proposition $B$ sur les hypothèses contenues dans $B$. Formellement :

$$\begin{array}{rcccl} \mathbf{BetP}\{m^\Omega\} & : & \Omega & \rightarrow & [0,1] \\ & & \omega_k & \mapsto & \mathbf{BetP}\{m^\Omega\}(\omega_k) \end{array} \tag{2.14}$$

où $\mathbf{BetP}\{m^\Omega\}(\omega_k), \forall \omega_k \in \Omega$ est donnée par :

$$\mathbf{BetP}\{m^\Omega\}(\omega_k) = \frac{1}{(1-m^\Omega(\emptyset))} \sum_{B \subseteq \Omega, \omega_k \in B} \frac{m^\Omega(B)}{|B|} \tag{2.15}$$

Les différentes propriétés de cette transformation sont analysées dans [Sme05c].

La décision est généralement prise en choisissant l'élément $\omega_k$ possédant la plus grande probabilité pignistique :

$$\omega_0 = \underset{\omega_k \in \Omega}{\operatorname{argmax}} \ \mathbf{BetP}\{m^\Omega\}(\omega_k) \tag{2.16}$$

**Exemple 3.** *La transformée pignistique de la BBA donnée dans la* $6^e$ *colonne du tableau 2.1 fournit la distribution pignistique suivante :* $\mathbf{BetP}\{m^\Omega\}(\omega_1) \approx 0.28$, $\mathbf{BetP}\{m^\Omega\}(\omega_2) \approx 0.29$ *et* $\mathbf{BetP}\{m^\Omega\}(\omega_3) \approx 0.43$. *La décision sur l'état du système* $\omega_0$ *pourrait être* $\omega_0 = \omega_3$ *signifiant que l'état réel du système est l'état* $\omega_3$.

#### 2.1.5.2 Utilisation des plausibilités pour la prise de décision

L'agent qui se base sur la transformée pignistique lors de la phase de prise de décision a un comportement rationnel en maximisant l'*utilité espérée*. Alternativement, les *plausibilités* et les *crédibilités* peuvent être utilisées lorsque l'agent a une attitude plutôt *optimiste* ou *pessimiste*.

Dans le cas des plausibilités, on dit que l'agent a une attitude optimiste dans le sens où la plausibilité d'un élément représente la valeur maximale de la masse de croyance sur cet élément qui pourrait être atteinte si des informations supplémentaires parvenaient au système de fusion (ce dernier utilisant la combinaison conjonctive pour intégrer cette nouvelle information). La distribution de probabilités sur les singletons à partir de laquelle une décision peut être prise est alors donnée $\forall \omega_k \in \Omega$ par :

$$\mathbf{PlP}\{m^\Omega\}(\omega_k) \quad = \quad \frac{pl^\Omega(\omega_k)}{\sum\limits_{\omega_k \in \Omega} pl^\Omega(\omega_k)} \tag{2.17}$$

Notons que la plausibilité d'un singleton $pl^\Omega(\omega_k)$ est égale à sa communalité $q^\Omega(\omega_k)$. De la même manière, un critère adapté à un agent « pessimiste » peut être obtenue en utilisant les crédibilités (la crédibilité d'un singleton $bel^\Omega(\omega_k)$ est simplement égal à sa masse après normalisation de la BBA $m^\Omega$). On peut remarquer que pour tout élément $A \in 2^\Omega$ on a :

$$bel^\Omega(A) \leq \mathbf{BetP}\{m^\Omega\}(A) \leq pl^\Omega(A) \tag{2.18}$$

et en particulier lorsque les plausibilités et les crédibilités sont calculées sur les singletons.

**Remarque 2.2.** *Le cadre de pari utilisé lors de la décision peut être différent du cadre de discernement $\Omega$ de la BBA [Sme94b, Sme02b, Sme00a, Mer06] et le* coût *d'une décision peut être intégré dans la prise de décision (voir [Mer06] pour un exemple d'application dans le cadre de la reconnaissance d'adresses postales).*

### 2.1.6 Gestion du conflit

Lors de la combinaison par la CRC de deux BBA $m_1^\Omega$ et $m_2^\Omega$, l'intersection entre les éléments focaux peut être vide et dans ce cas la BBA résultante a une masse non nulle sur l'ensemble vide, i.e. $m_{1 \circledcirc 2}^\Omega(\emptyset) > 0$. Cette valeur est appelée *conflit* et quantifie la discordance entre les sources de croyance. Il est nécessaire d'analyser ce conflit pour deux raisons majeures :

- L'ensemble vide étant absorbant par l'opérateur CRC, la combinaison avec une troisième BBA $m_3^\Omega$ ne fera qu'augmenter la valeur du conflit, i.e. $m_{1 \circledcirc 2 \circledcirc 3}^\Omega(\emptyset) \geq m_{1 \circledcirc 2}^\Omega(\emptyset)$.
- Lors de la phase de décision, la fonction de croyance est transformée en distribution de probabilités pignistiques sur les singletons. Or, cette distribution est normalisée par l'opposé du conflit (Eq. 2.16) mais si le conflit est élevé, quel est le sens de la distribution de probabilités obtenue ?

Il est nécessaire d'analyser les origines du conflit qui peuvent être multiples [Sme05a]. Une des origines est la non exhaustivité du cadre de discernement. Ainsi, la quantité de masse de conflit peut être interprétée comme la croyance sur une nouvelle hypothèse [DS04]. On parle alors de « TBM en monde ouvert ».

Une autre raison pouvant expliquer le conflit est le non respect des hypothèses sous jacentes à l'application de la règle de combinaison conjonctive employée. Par exemple, la CRC (Eq. 2.9)

suppose que les BBA sont distinctes [Sme93, YSM01, YSM02a, YSM02b] or si ce n'est pas le cas il est préférable d'utiliser la règle prudente proposée récemment par Denœux [Den07].

Une troisième origine possible du conflit provient de la nature de la combinaison conjonctive employée et de celle des BBA. Par exemple, la CRC n'est pas idempotente et par conséquent la combinaison d'une BBA avec elle-même modifie la BBA initiale et peut induire du conflit. Ce conflit est alors appelé *conflit interne*. Pour éviter ce phénomène, deux solutions existent. La première consiste à construire des distributions de masses consonantes en s'inspirant des sous-ensembles flous comme dans [HCR05, GCBR05a, RPRP07]. La seconde consiste à changer la règle de combinaison conjonctive et à utiliser par exemple une règle minimisant le conflit [Cat03].

Le conflit peut être aussi interprété comme une véritable source d'information par exemple pour regrouper des fonctions de croyance [Sch04, DM04], associer des objets [AS01], détecter de nouveaux objets [DS04], suivre des cibles [RS06] ou détecter des contours dans les images [CCFM04]. Nous l'avons exploité pour détecter des changements d'états et filtrer des croyances évoluant au cours du temps [RRP07c] ainsi que pour reconnaître des séquences d'états [RPR06, RRP07a, RRP07b].

Dans certaines applications, il peut être intéressant de traiter le conflit automatiquement et pour cela il existe des règles dites de *redistribution* [LCV02]. Celle de Yager par exemple transfère le conflit sur le doute total ($\Omega$) tandis que la règle de Dubois et Prade, aussi appelée règle *adaptative*, transfère le conflit sur l'union des propositions impliquées.

## 2.2 Opérations avancées

Dans cette section, nous présentons des outils du TBM permettant de traiter des problèmes de représentation des connaissances et de combinaison complexes. Plus précisément, des éléments de réponse aux questions suivantes sont apportés :

- Comment traiter des problèmes de *diagnostic* ou de *prédiction* ?
- Comment combiner des fonctions de croyance dont les *cadres de discernement* sont *différents* ?
- Comment combiner des distributions de *probabilités* avec des fonctions de *croyances* ?
- Comment combiner des fonctions de croyance dont les sources ne sont *pas indépendantes* ?

### 2.2.1 Le Théorème de Bayes Généralisé

Nous décrivons tout d'abord le rôle et le principe du Théorème de Bayes sur un exemple concernant la prédiction météorologique (voir aussi [Sme93] pour un exemple concernant le diagnostic médical et [DS04] pour l'identification de cibles). Nous décrivons ensuite le Théorème de Bayes Généralisé (GBT : *Generalized Bayesian Theorem*) et donnons quelques exemples d'application.

#### 2.2.1.1 Le principe sur un exemple : problème de prédiction du temps

Soit un problème de *prédiction météorologique*. On note $\Omega_j = \{E_j, P_j, N_j\}$ l'espace de discernement concernant le temps au jour $j$ signifiant que le temps est soit ensoleillé ($E_j$) soit pluvieux ($P_j$) ou soit nuageux ($N_j$).

Une *base de connaissance* liant le temps du jour précédent $j-1$ au temps du jour courant $j$ a été établie suite à des campagnes de mesures. La base est décrite dans le tableau 2.4 : chaque ligne décrit la croyance sur le temps au jour $j$ conditionnellement au temps du jour précédent $j-1$. Ces

croyances sont définies sur $\Omega_j$ conditionnellement aux singletons du cadre de discernement $\Omega_{j-1}$ et sont notées $m^{\Omega_j}[E_{j-1}]$, $m^{\Omega_j}[P_{j-1}]$ et $m^{\Omega_j}[N_{j-1}]$ (une ligne par BBA conditionnelle).

| | $\emptyset_j$ | $\{E_j\}$ | $\{P_j\}$ | $\{E_j, P_j\}$ | $\{N_j\}$ | $\{E_j, N_j\}$ | $\{P_j, N_j\}$ | $\Omega_j$ |
|---|---|---|---|---|---|---|---|---|
| $[E_{j-1}]$ | 0 | 0.59 | 0.01 | 0.03 | 0.07 | 0.21 | 0.01 | 0.08 |
| $[P_{j-1}]$ | 0 | 0.06 | 0.31 | 0.02 | 0.45 | 0.11 | 0.02 | 0.03 |
| $[N_{j-1}]$ | 0 | 0.14 | 0.12 | 0.04 | 0.63 | 0.01 | 0.05 | 0.01 |

**TAB. 2.4 :** *Base de connaissance liant l'état du temps au jour $j$ connaissant l'état du temps au jour $j-1$.*

A partir de cette base deux problèmes seront traités :

1. *Raisonnement par inférence ascendante* ou *diagnostic :* Disposant d'une BBA $m^{\Omega_j}$ traduisant le temps au jour $j$, comment déterminer le temps au jour précédent $j-1$, i.e. comment calculer la BBA $m^{\Omega_{j-1}}$ ? On s'intéresse aux *causes* d'une observation.
2. *Raisonnement par inférence descendante* ou *prédiction :* Disposant d'une BBA $m^{\Omega_j}$ traduisant le temps au jour $j$, comment *prédire* la BBA sur le temps du lendemain, i.e. comment calculer la BBA $m^{\Omega_{j+1}}$ ? On s'intéresse aux *effets* d'une hypothèse sur une base de connaissance.

Les solutions sont apportées par des mécanismes d'*inférence* dont le plus connu est le théorème de Bayes dans le cadre de la théorie des probabilités. Ce théorème a été généralisé aux fonctions de croyance (GBT) et on parle alors d'*inférence évidentielle*[3].

#### 2.2.1.2 La solution probabiliste : le théorème de Bayes

On appelle $\Theta = \{\theta_1, \ldots, \theta_i, \ldots\}$ *l'espace des hypothèses* et $\Omega = \{\omega_1, \ldots, \omega_k, \ldots\}$ *l'espace des observations*[4]. En théorie des probabilités, la relation entre les observations $\omega_k \in \Omega$ et les hypothèses $\theta_i \in \Theta$ est représentée par un jeu de probabilités conditionnelles donnant la distribution de probabilités sur les observations $\omega_k \in \Omega$ sachant chaque hypothèse $\theta_i \in \Theta$, i.e. $P^{\Omega}(.|\theta_i)$. Les systèmes d'inférence probabilistes décrit dans ce paragraphe sont à la base de nombreux systèmes actuels, par exemple pour le traitement de signaux audio [Rab89] et de vidéo [Mur02].

**Raisonnement par inférence ascendante *ou* diagnostic** Suite à l'observation d'un système, une source d'information fournit son *opinion sur les valeurs des hypothèses* $\theta_i$ dans $\Theta$ (concernant l'état du système par exemple) sous la forme d'une vraisemblance définie sur $\Omega$ conditionnellement à l'hypothèse $\theta_i \in \Theta$ et que l'on note $P(\omega_k|\theta_i)$. Il est alors possible de remonter aux causes de cette observation en estimant les distributions de probabilités a posteriori $P^{\Theta}(.|\omega_k)$ par le *théorème de Bayes* :

$$P(\theta_i|\omega_k) = \frac{P(\theta_i) \cdot P(\omega_k|\theta_i)}{\sum_{\theta_i \in \Theta} P(\theta_i) \cdot P(\omega_k|\theta_i)}, \forall \theta_i \in \Theta \tag{2.19}$$

ou de façon moins formelle :

$$\text{a posteriori sur } \Theta \propto \begin{array}{c}\text{a priori}\\ \text{sur les hypothèses}\end{array} \times \begin{array}{c}\text{vraisemblance des observations}\\ \text{conditionnelle aux hypothèses}\end{array} \tag{2.20}$$

[3]Le terme « évidentiel » est utilisé pour qualifier des mécanismes basés sur les fonctions de croyance.

[4]Nous préférons utiliser $\Theta$ et $\Omega$ au lieu de $\Omega_{j-1}$ et $\Omega_j$ pour définir les outils dans un cadre général.

Notons que le symbole $\propto$ signifie « proportionnel à », sous-entendu à une constante près assurant que la somme des éléments est unitaire. Ce symbole sera utilisé à plusieurs reprises dans ce manuscrit.

**Exemple 4.** Problème de diagnostic météorologique : *on cherche à connaître les causes du temps au jour courant $j$ disposant d'une base de connaissance liant le temps au jour $j$ et $j-1$. Autrement dit, on souhaite déterminer le temps du jour précédent $j-1$ pouvant avoir causé le temps au jour courant $j$. Pour cela, on cherche à estimer les distributions conditionnelles $P^{\Omega_{j-1}}(.|\omega_j)$ en utilisant le théorème de Bayes. Notons que $\Omega_j$ ($\equiv \Omega$) est l'espace des observations et que $\Omega_{j-1}$ ($\equiv \Theta$) est celui des hypothèses.*

*Comme on ne dispose initialement que d'une base de connaissance sous la forme de fonctions de croyance (tableau 2.4) et que l'on emploie des outils probabilistes, il est nécessaire de transformer les BBA conditionnelles $m^{\Omega_j}[\omega_{j-1}]$ (Tab. 2.4) en probabilités conditionnelles $P^{\Omega_j}(.|\omega_{j-1})$ en utilisant la transformation pignistique (Eq. 2.16). Les résultats sont donnés dans le tableau 2.5. Ne disposant d'aucune connaissance a priori sur le temps au jour précédent $j-1$, la distribution de probabilités a priori est donc uniforme, i.e. $P(\omega_{j-1}) = 1/3, \forall \omega_{j-1} \in \Omega_{j-1}$. On dispose ainsi des éléments pour appliquer le théorème de Bayes.*

| | $\{E_j\}$ | $\{P_j\}$ | $\{N_j\}$ |
|---|---|---|---|
| $[E_{j-1}]$ | 0.74 | 0.05 | 0.21 |
| $[P_{j-1}]$ | 0.13 | 0.34 | 0.53 |
| $[N_{j-1}]$ | 0.17 | 0.17 | 0.66 |

**TAB. 2.5 :** *Base de connaissance probabiliste $P^{\Omega_j}(.|\omega_{j-1})$ issue de la transformation pignistique des distributions du tableau 2.4.*

*Les distributions a posteriori obtenues par le théorème de Bayes figurent dans le tableau 2.7a. Elles ont logiquement les mêmes valeurs que les vraisemblances du tableau 2.5 (en inversant les conditionnements) : n'ayant aucune connaissance a priori sur le temps à $j-1$, la base de connaissance n'est donc pas modifiée a posteriori. Notons que les valeurs en gras représentent les maxima de probabilités pour chaque conditionnement.*

**Raisonnement par inférence descendante *ou* prédiction** Disposant d'une *base de connaissances* $P^{\Omega}(.|\theta_i)$ donnant la probabilité d'apparition d'une observation conditionnellement aux hypothèses, il est possible d'estimer la distribution sur les observations $P(\omega_k)$ suite à une hypothèse émise $P(\theta_i)$ à l'aide de l'équation de conditionnement :

$$P(\omega_k) = \sum_{\theta_i \in \Theta} P(\theta_i) \cdot P(\omega_k|\theta_i) \tag{2.21}$$

Cette équation permet d'estimer les *effets* d'une hypothèse sur les observations et elle est classiquement utilisée pour les problèmes de prédiction [Mur02].

**Exemple 5.** Problème de prédiction météorologique : *on souhaite déterminer le temps au jour $j+1$, i.e. la distribution de probabilités $P^{\Omega_{j+1}}$. On suppose dans un premier temps n'avoir aucune information a priori sur $j$ et on fait donc appel au principe d'équiprobabilité permettant de traduire cette*

*méconnaissance par une distribution uniforme :* $P^{\Omega_j}(E_j) = P^{\Omega_j}(P_j) = P^{\Omega_j}(N_j) = 1/3$. *On utilise la base de connaissance probabiliste du tableau 2.5 donnant la relation entre le temps du jour* $j-1$ *et celui du lendemain* $j$. *On dispose donc des éléments pour appliquer l'équation 2.21 générant une prédiction probabiliste dont le résultat est donné à la première ligne de l'équation 2.22 :*

$$P^{\Omega_{j+1}}(E_{j+1}) = 0.35 \quad P^{\Omega_{j+1}}(P_{j+1}) = 0.19 \quad P^{\Omega_{j+1}}(N_{j+1}) = 0.46 \tag{2.22}$$

*par exemple* $P^{\Omega_{j+1}}(E_{j+1}) = \sum_{\omega_j \in \Omega_j} P(\omega_j) \cdot P(\omega_{j+1}|\omega_j) = 1/3 \cdot 0.74 + 1/3 \cdot 0.13 + 1/3 \cdot 0.17 \approx 0.35$. *Cette distribution montre que le temps prédit est probablement* nuageux.

*On suppose à présent qu'un a priori sur le temps à* $j$ *existe : on sait que le temps n'est* pas ensoleillé. *En terme de probabilités, on dispose d'une distribution a priori de la forme* $P^{\Omega_j}(\{P_j\}) = P^{\Omega_j}(\{N_j\}) = 1/2$ *et on obtient :*

$$P^{\Omega_{j+1}}(\{E_{j+1}\}) = 0.15 \quad P^{\Omega_{j+1}}(\{P_{j+1}\}) = 0.25 \quad P^{\Omega_{j+1}}(\{N_{j+1}\}) = 0.60 \tag{2.23}$$

#### 2.2.1.3 La solution TBM : le Théorème de Bayes Généralisé

Smets a proposé une généralisation du théorème de Bayes dans le cadre du TBM qu'il a nommé le Théorème de Bayes Généralisé (GBT) [Sme78, Sme93] (voir aussi les travaux d'Appriou [App91]). Par rapport à sa version probabiliste, le GBT est capable de manipuler des fonctions de croyance.

Outre le fait que les fonctions de croyance permettent une modélisation plus fine de la connaissance et qu'elles généralisent les probabilités et les possibilités, elles permettent de modéliser la *méconnaissance* sur l'a priori nécessaire au calcul de la distribution a posteriori.

**Le Théorème de Bayes Généralisé** A la différence du théorème de Bayes qui génère une distribution de probabilités a posteriori, le GBT génère une distribution de croyances pouvant prendre la forme de plausibilités, de communalités, de masses...

Les expressions analytiques des distributions générées par le GBT ont été dérivées par Smets [Sme93]. Parmi les plus utilisées, trois sont basées sur les plausibilités $pl^{\Omega}[\theta_i]$ définies conditionnellement aux *singletons* $\theta_i \in \Theta$ réduisant ainsi le coût de stockage en mémoire lors de l'implémentation[5]. Les expressions analytiques sont basées sur l'hypothèse que l'a priori sur $\Theta$ est vide. Le GBT génère alors les plausibilités, communalités et masses a posteriori suivantes :

$$pl^{\Theta}[\omega](\theta) = 1 - \prod_{\theta_i \in \theta} \Big(1 - pl^{\Omega}[\theta_i](\omega)\Big) \tag{2.24a}$$

$$q^{\Theta}[\omega](\theta) = \prod_{\theta_i \in \theta} pl^{\Omega}[\theta_i](\omega) \tag{2.24b}$$

$$m^{\Theta}[\omega](\theta) = \prod_{\theta_i \in \theta} pl^{\Omega}[\theta_i](\omega) \cdot \prod_{\theta_i \in \overline{\theta}} \Big(1 - pl^{\Omega}[\theta_i](\omega)\Big) \tag{2.24c}$$

**Exemple 6.** Problème de diagnostic météorologique : *Nous appliquons le GBT pour déterminer la BBA sur le temps à* $j-1$ *conditionnellement au temps à* $j$ *sans a priori sur* $\Omega_{j-1}$. *Les résultats du GBT sont obtenus à partir des données du tableau 2.4 en trois étapes :*

[5] Le coût mémoire d'une base de connaissance probabiliste $P^{\Omega}(.|\theta_i)$ est de $|\Omega| \times |\Theta|$ tandis que pour une base crédibiliste, elle est de $2^{|\Omega|} \times |\Theta|$ pour les plausibilités et $2^{|\Omega|} \times 2^{|\Theta|}$ pour les masses.

1. *Transformation des BBA conditionnelles $m^{\Omega_j}[\omega_{j-1}]$ du tableau 2.4 en plausibilités conditionnelles $pl^{\Omega_j}[\omega_{j-1}]$ en utilisant l'équation 2.4a,*
2. *Application du GBT (Eq. 2.24c) :*

$$m^{\Omega_{j-1}}[\omega_j](\omega) = \prod_{\omega_{j-1}\in\omega} pl^{\Omega_j}[\omega_{j-1}](\omega_j) \cdot \prod_{\omega_{j-1}\in\overline{\omega}} \Big(1 - pl^{\Omega_j}[\omega_{j-1}](\omega_j)\Big)$$

*Les plausibilités $pl^{\Omega_{j-1}}[B], \forall B \subseteq \Omega_j$ sont explicitées dans le tableau 2.6.*

| | $\emptyset_{j-1}$ | $\{E_{j-1}\}$ | $\{P_{j-1}\}$ | $\{E_{j-1}, P_{j-1}\}$ | $\{N_{j-1}\}$ | $\{E_{j-1}, N_{j-1}\}$ | $\{P_{j-1}, N_{j-1}\}$ | $\Omega_{j-1}$ |
|---|---|---|---|---|---|---|---|---|
| $[\emptyset_j]$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $[E_j]$ | 0 | 0.910 | 0.220 | 0.930 | 0.200 | 0.928 | 0.376 | 0.944 |
| $[P_j]$ | 0 | 0.130 | 0.380 | 0.461 | 0.220 | 0.321 | 0.516 | 0.579 |
| $[\{E_j, P_j\}]$ | 0 | 0.930 | 0.550 | 0.969 | 0.370 | 0.956 | 0.717 | 0.980 |
| $[N_j]$ | 0 | 0.370 | 0.610 | 0.754 | 0.700 | 0.811 | 0.883 | 0.926 |
| $[\{E_j, N_j\}]$ | 0 | 0.990 | 0.690 | 0.997 | 0.880 | 0.999 | 0.963 | 1 |
| $[\{P_j, N_j\}]$ | 0 | 0.410 | 0.940 | 0.965 | 0.860 | 0.917 | 0.992 | 0.995 |
| $[\Omega_j]$ | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**TAB. 2.6 :** *Les plausibilités a posteriori $pl^{\Omega_{j-1}}[\omega_j]$ définies sur $\Omega_{j-1}$ conditionnellement aux hypothèses et sous-ensembles d'hypothèses de $\Omega_j$ obtenues par le GBT.*

3. *Application de la transformation pignistique. La distribution de probabilités pignistiques associée à la BBA $m^{\Omega_{j-1}}[E_j]$ est donnée à la première ligne du tableau 2.7b et montre que le temps au jour $j-1$ était probablement ensoleillé avec une pignistique de $0.77$.*

*La distribution de probabilités pignistiques conditionnelles obtenues par le GBT (Tab. 2.7b) est différente de la distribution de probabilités conditionnelles obtenue par le théorème de Bayes (Tab. 2.7a). En particulier, les probabilités maximales pour chaque conditionnement sont différentes et surtout les valeurs maximales ne concernent pas les mêmes singletons*[6].

| | $\{E_{j-1}\}$ | $\{P_{j-1}\}$ | $\{N_{j-1}\}$ |
|---|---|---|---|
| $[E_j]$ | **0.74** | 0.05 | 0.21 |
| $[P_j]$ | 0.13 | 0.34 | **0.53** |
| $[N_j]$ | 0.17 | 0.17 | **0.66** |

*a) Probabilité a posteriori après application du théorème de Bayes avec a priori uniforme.*

| | $\{E_{j-1}\}$ | $\{P_{j-1}\}$ | $\{N_{j-1}\}$ |
|---|---|---|---|
| $[E_j]$ | **0.77** | 0.12 | 0.11 |
| $[P_j]$ | 0.16 | **0.55** | 0.29 |
| $[N_j]$ | 0.19 | 0.36 | **0.44** |

*b) Probabilités pignistiques a posteriori après application du GBT avec a priori vide.*

**TAB. 2.7 :** *Distributions de probabilités $P^{\Omega_{j-1}}(.|\omega_j)$ au jour $j-1$ conditionnellement aux éléments de $\Omega_j$.*

*La différence vient du fait que la base de connaissance a été transformée en probabilités et par conséquent, une connaissance a priori forte a été ajoutée.*

[6]Notons que, si au lieu de transformer les BBA conditionnelles en pignistique on choisit plutôt de les transformer en plausibilités normalisées **PlP** (Eq. 2.17), les résultats numériques sont différents mais les maxima concernent les mêmes singletons que dans le cas pignistique.

**Dualité GBT – DRC** La relation $pl^{\Omega}[\theta_1](\theta_2) = pl^{\Omega}(\theta_1 \cap \theta_2), \forall \theta_1 \subseteq \Theta, \theta_2 \subseteq \Theta$ (avec $pl^{\Omega}(\theta_1) = 1$) [Sha76] permet d'écrire $pl^{\Omega}[\theta](\omega) = pl^{\Omega\times\Theta}(\omega \times \theta)$. Par conséquent, l'égalité suivante est vérifiée :

$$pl^{\Omega}[\theta](\omega) = pl^{\Theta}[\omega](\theta) \tag{2.25}$$

Le couplage de l'égalité 2.25 avec l'équation 2.24a permet d'obtenir une expression de la DRC basée sur les plausibilités [Sme93]. Celle-ci est fréquemment utilisée car elle minimise le stockage mémoire et les coûts de calcul par rapport à la DRC classique (Eq. 2.12) :

$$pl^{\Omega}[\theta](\omega) \;=\; 1 - \prod_{\theta_i \in \theta} \left(1 - pl^{\Omega}[\theta_i](\omega)\right) \tag{2.26}$$

**Prise en compte d'a priori sur les hypothèses et prédiction** Lorsqu'un a priori $m_o^{\Theta}$ existe sur $\Theta$ (les hypothèses), pouvant être vide ou prendre toute autre forme (cf. exemples de BBA dans le tableau 2.1), il est pris en compte en utilisant la forme conjonctive suivante [Sme93] :

$$pl^{\Omega}(\omega) \;=\; \sum_{\theta \subseteq \Theta} m_o^{\Theta}(\theta) \cdot pl^{\Omega}[\theta](\omega), \forall \omega \subseteq \Omega \tag{2.27}$$

où $pl^{\Omega}[\theta](\omega)$ est donnée par l'équation 2.26. L'équation 2.27, qui est analogue à l'équation 2.21 dans le cadre des probabilités, est généralisable aux fonctions de croyance quelconques (fonctions de masses $m$, communalités $q\ldots$) et que l'on note $f^{\Omega} \in \{m^{\Omega}, pl^{\Omega}, bel^{\Omega}, b^{\Omega}, q^{\Omega}\}$. L'expression généralisant l'équation 2.27 est donc [Sme93] :

$$f^{\Omega}(\omega) \;=\; \sum_{\theta \subseteq \Theta} m_o^{\Theta}(\theta) \cdot f^{\Omega}[\theta](\omega), \forall \omega \subseteq \Omega \tag{2.28}$$

Comme le montre l'équation 2.28, il est nécessaire de disposer des croyances définies sur $\Theta$ conditionnellement aux sous-ensembles $\omega = \bigcup_k \omega_k \subseteq \Omega$. Comme cela a été précisé par Smets [Sme93], on ne dispose généralement que des croyances définies conditionnellement aux singletons car elles sont plus simples à déterminer. Il s'agit alors de calculer les croyances définies conditionnellement aux sous-ensembles d'hypothèses à l'aide d'une règle disjonctive. La DRC basée sur les plausibilités a été décrite précédemment mais il est aussi possible d'utiliser une DRC sur les BBA conditionnelles :

$$m^{\Omega}[\theta] \;=\; \bigcirc\!\!\!\!\bigcup_{\theta_i \in \theta} m^{\Omega}[\theta_i], \forall \theta \subseteq \Theta \tag{2.29}$$

Par exemple la BBA $m^{\Omega_j}[\{E_{j-1}, P_{j-1}\}]$ est obtenue par l'application de la DRC sur $m^{\Omega_j}[E_{j-1}]$ et $m^{\Omega_j}[P_{j-1}]$ en supposant ces BBA distinctes. Le résultat est donné à la quatrième ligne du tableau 2.8.

**Exemple 7.** Problème de prédiction météorologique : *On souhaite prédire le temps au jour $j+1$ à partir de la base de connaissances et des données a priori. Contrairement à la prédiction probabiliste de l'exemple 5, nous disposons des BBA conditionnelles nécessaires à l'application du GBT (Tab. 2.4).*

*On applique ensuite l'équation 2.28 (avec $f \equiv m$) pour prédire la BBA recherchée. Pour appliquer cette équation, nous devons déterminer les BBA $m^{\Omega_j}[\omega_{j-1}]$ définies sur $\Omega_j$ conditionnellement aux hypothèses et sous-ensembles d'hypothèses de $\Omega_{j-1}$. Pour cela, nous avons deux solutions :*

- *Appliquer la DRC avec les plausibilités : ceci nécessite de transformer les BBA conditionnelles du tableau 2.4 en plausibilités puis d'appliquer l'équation 2.26 et enfin de transformer les plausibilités conditionnelles obtenues en BBA (Eq. 2.4b),*

| | $\emptyset_j$ | $\{E_j\}$ | $\{P_j\}$ | $\{E_j, P_j\}$ | $\{N_j\}$ | $\{E_j, N_j\}$ | $\{P_j, N_j\}$ | $\Omega_j$ |
|---|---|---|---|---|---|---|---|---|
| $[\emptyset_{j-1}]$ | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $[E_{j-1}]$ | 0 | 0.590 | 0.010 | 0.030 | 0.070 | 0.210 | 0.010 | 0.080 |
| $[P_{j-1}]$ | 0 | 0.060 | 0.310 | 0.020 | 0.450 | 0.110 | 0.020 | 0.030 |
| $[\{E_{j-1}, P_{j-1}\}]$ | 0 | 0.035 | 0.003 | 0.207 | 0.032 | 0.472 | 0.036 | 0.215 |
| $[N_{j-1}]$ | 0 | 0.140 | 0.120 | 0.040 | 0.630 | 0.010 | 0.050 | 0.010 |
| $[\{E_{j-1}, N_{j-1}\}]$ | 0 | 0.083 | 0.001 | 0.105 | 0.044 | 0.552 | 0.027 | 0.188 |
| $[\{P_{j-1}, N_{j-1}\}]$ | 0 | 0.008 | 0.037 | 0.071 | 0.283 | 0.192 | 0.303 | 0.105 |
| $[\Omega_{j-1}]$ | 0 | 0.005 | 0 | 0.068 | 0.020 | 0.396 | 0.036 | 0.475 |

**TAB. 2.8 :** *Les masses définies sur $\Omega_j$ conditionnellement aux hypothèses et sous-ensembles d'hypothèses de $\Omega_{j-1}$ obtenues par la DRC.*

- *Ou appliquer la DRC définie à l'équation 2.29 directement sur les BBA conditionnelles du tableau 2.4.*

*Les deux solutions mènent aux BBA du tableau 2.8.*

*Dans un premier temps, on suppose que l'a priori sur $\Omega_j$ est vide (on ne connaît rien du temps actuel). L'équation 2.28 est donc appliquée avec $m_o^{\Omega_j}(\Omega_j) = 1$ et en utilisant les BBA conditionnelles obtenues précédemment (Tab. 2.8). La BBA prédite est alors la suivante :*

$$\begin{array}{rcl@{\qquad}rcl} m^{\Omega_{j+1}}(\emptyset_{j+1}) & = & 0 & m^{\Omega_{j+1}}(\{E_{j+1}\}) & = & 0.005 \\ m^{\Omega_{j+1}}(\{P_{j+1}\}) & = & 0 & m^{\Omega_{j+1}}(\{E_{j+1}, P_{j+1}\}) & = & 0.068 \\ m^{\Omega_{j+1}}(\{N_{j+1}\}) & = & 0.02 & m^{\Omega_{j+1}}(\{E_{j+1}, N_{j+1}\}) & = & 0.396 \\ m^{\Omega_{j+1}}(\{P_{j+1}, N_{j+1}\}) & = & 0.036 & m^{\Omega_{j+1}}(\Omega_j) & = & 0.475 \end{array} \tag{2.30}$$

*dont la distribution de probabilité pignistique est donnée par :*

$$\mathbf{BetP}\{m^{\Omega_{j+1}}\}(E_{j+1}) \approx 0.40 \quad \mathbf{BetP}\{m^{\Omega_{j+1}}\}(P_{j+1}) \approx 0.21 \quad \mathbf{BetP}\{m^{\Omega_{j+1}}\}(N_{j+1}) \approx 0.39 \tag{2.31}$$

*Cette distribution indique que le temps prédit est soit* ensoleillé *soit* nuageux *(les valeurs des probabilités sont trop proches pour départager les deux hypothèses). Les valeurs obtenues sont sensiblement différentes du cas probabiliste (Ex. 2.22) du fait de la modification de la base de connaissance pour appliquer le Théorème de Bayes.*

*Dans un deuxième temps, on modifie l'a priori : on sait que le temps actuel à $j$ n'est* pas ensoleillé. *L'équation 2.28 est appliquée avec $m_o^{\Omega_j}(\{P_j, N_j\}) = 1$ et en utilisant les BBA conditionnelles du tableau 2.8. La BBA prédite est alors la suivante :*

$$\begin{array}{rcl@{\qquad}rcl} m^{\Omega_{j+1}}(\emptyset_{j+1}) & = & 0 & m^{\Omega_{j+1}}(\{E_{j+1}\}) & = & 0.008 \\ m^{\Omega_{j+1}}(\{P_{j+1}\}) & = & 0.037 & m^{\Omega_{j+1}}(\{E_{j+1}, P_{j+1}\}) & = & 0.071 \\ m^{\Omega_{j+1}}(\{N_{j+1}\}) & = & 0.284 & m^{\Omega_{j+1}}(\{E_{j+1}, N_{j+1}\}) & = & 0.192 \\ m^{\Omega_{j+1}}(\{P_{j+1}, N_{j+1}\}) & = & 0.303 & m^{\Omega_{j+1}}(\Omega_j) & = & 0.105 \end{array} \tag{2.32}$$

*dont la distribution de probabilité pignistique est donnée par :*

$$\mathbf{BetP}\{m^{\Omega_{j+1}}\}(E_{j+1}) \approx 0.17 \quad \mathbf{BetP}\{m^{\Omega_{j+1}}\}(P_{j+1}) \approx 0.26 \quad \mathbf{BetP}\{m^{\Omega_{j+1}}\}(N_{j+1}) \approx 0.57 \tag{2.33}$$

*Cette distribution indique que le temps du lendemain est probablement* nuageux. *On peut remarquer que les valeurs sont assez proches du cas probabiliste 2.23 car on a ajouté un a priori.*

**GBT et DRC pour la propagation de croyances dans les réseaux évidentiels** Le GBT et la DRC sont au cœur des réseaux évidentiels [Sme93, XS94, YSM03] généralisant les réseaux Bayésiens [Mur02]. Un réseau est un modèle graphique dont l'objectif est de représenter les connaissances sous forme de graphes orientés mettant en évidence la causalité entre des variables d'un système étudié.

L'équation 2.28 est représentée sur la figure 2.3 sous la forme d'un modèle graphique évidentiel. Cette représentation montre un graphe orienté à deux variables prenant leurs valeurs discrètes sur $\Theta$ et $\Omega$. L'arc orienté partant du nœud $\Theta$ et allant vers le nœud de $\Omega$ signifie que « $\theta \subseteq \Theta$ cause $\omega \subseteq \Omega$ » (sous l'hypothèse $\theta$, on observe $\omega$) et donc il existe un lien de causalité (dépendance) entre les deux variables.

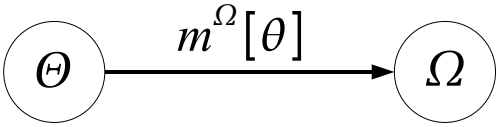



**FIG. 2.3 :** *Modèle graphique représentant le lien de causalité entre deux variables à valeurs discrètes dans $\Theta$ et $\Omega$. Le lien de causalité est traduit par un conditionnement ($m^{\Omega}[\theta], \forall\theta \subseteq \Theta$).*

Si un ou plusieurs nœuds d'un réseau reçoit une BBA (instanciation, par exemple $\Theta$ reçoit $m^{\Theta}$), on peut calculer son influence sur l'ensemble du réseau grâce à des mécanismes de *propagation de croyance*. Pour illustrer ces mécanismes, on considère le réseau très simple de la figure 2.3 avec deux variables où l'arc orienté, qui représente les liens de causalité, est caractérisé par un jeu de plausibilités conditionnelles $pl^{\Omega}[\theta_i], \forall\theta_i \in \Theta$ :

**Propagation forward :** si le nœud $\Theta$ reçoit une BBA $m^{\Theta}$, cette dernière est *propagée* vers $\Omega$ (en suivant la flèche d'où le qualificatif *forward*) par l'équation 2.27 qui calcule les plausibilités $pl^{\Omega}[\theta]$ conditionnelles aux sous-ensembles $\theta \subseteq \Theta$ par la DRC et applique une règle conjonctive pour calculer l'impact sur $\Omega$.

**Propagation backward :** si le nœud $\Omega$ reçoit une BBA $m^{\Omega}$, cette dernière est *propagée* vers $\Theta$ (dans le sens contraire à celui indiqué par la flèche d'où le qualificatif *backward*) par une règle conjonctive similaire à 2.27 :

$$pl^{\Theta}(\theta) \;=\; \sum_{\omega\subseteq\Omega} m^{\Omega}(\omega)\cdot\left(1-\prod_{\omega_i\in\omega}\left(1-pl^{\Theta}[\omega_i](\theta)\right)\right), \forall\theta\subseteq\Theta \tag{2.34}$$

où les plausibilités $pl^{\Theta}[\omega_i]$ sont obtenues par le GBT (Eq. 2.24a).

**Propagation dans les deux directions :** si les deux nœuds reçoivent une BBA, il suffit d'appliquer la propagation *forward* avec $m^{\Theta}$, puis la propagation *backward* avec $m^{\Omega}$ et enfin de combiner les deux résultats avec la CRC.

Dans l'exemple de la prédiction météorologique, un modèle graphique peut être obtenu, similaire à celui de la figure 2.3, où le nœud $\Omega_j$ (le temps actuel) est représenté par $\Theta$ et le nœud $\Omega_{j+1}$ (le temps du jour suivant) par $\Omega$ explicitant le fait que la connaissance du temps au jour $j$ *influe* sur la prédiction du temps à $j+1$. Les mécanismes de propagation seront utilisés dans le chapitre 5.

### 2.2.2 Opérations sur les cadres de discernement

Dans le paragraphe 2.1.4, nous avons présenté des mécanismes de base de combinaison de fonctions de croyance dont le rôle est de prendre en compte toutes les informations disponibles afin d'améliorer la prise de décision. Ces mécanismes sont basés sur l'hypothèse que les fonctions de croyance sont définies sur le même espace de discernement mais cette contrainte limite les applications pratiques.

Le TBM propose des outils performants et adaptés lorsque les cadres de discernement sont différents. Afin d'illustrer l'utilisation de ces outils, nous considérons le cas de la combinaison de deux fonctions de croyance : la première est $m_1^{\Omega}$ définie sur un premier cadre $\Omega$, la seconde est $m_2^{\Theta}$ définie sur $\Theta$.

Le principe général de ces mécanismes consiste en trois étapes :

1. Construire le cadre de discernement commun,
2. Exprimer les croyances sur le cadre de discernement commun obtenu à l'étape 1. Les mécanismes sont basés sur le Principe du Minimum d'Information (PMI),
3. Combiner les fonctions de croyance obtenues à l'étape 2 par les règles de combinaison classiques.

Les deux mécanismes de base qui permettent le changement d'espace sont le *raffinement*, où l'espace devient plus *détaillé*, et le *grossissement*, où l'espace devient plus *grossier*.

#### 2.2.2.1 Raffinement et extension vide

**Raffinement** Un raffinement associe à chaque hypothèse $\omega_k \in \Omega$ une ou plusieurs hypothèses $\theta_i \in \Theta$. L'application $\rho : 2^{\Omega} \rightarrow 2^{\Theta}$ est un raffinement de $\Omega$ si et seulement si $\rho$ vérifie les contraintes suivantes :

- l'ensemble $\{\rho(\{\omega_k\}), \omega_k \in \Omega\} \subseteq 2^{\Theta}$ est une partition de $\Theta$,
- $\forall B \subseteq \Omega, \rho(B) = \bigcup_{\omega_k \in B} \rho(\{\omega_k\})$

L'image d'une union d'hypothèses est l'union des images par $\rho$ de chacune des hypothèses.

La figure 2.4 présente un exemple de raffinement où le cadre $\Theta$ est la version *raffinée* de $\Omega$. Les fléches partant de l'ensemble $\Omega$ et allant vers $\Theta$ décrivent le raffinement.

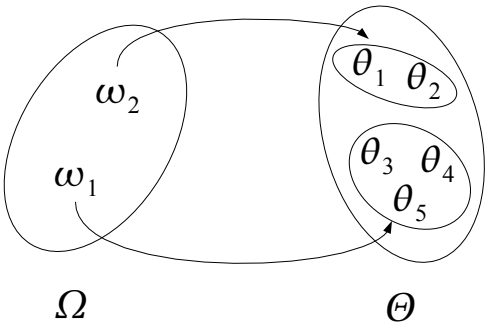



**FIG. 2.4 :** *Raffinement de cadre de discernement : $\Theta$ est un raffinement de $\Omega$.*

**Extension vide** Après le raffinement d'un cadre de discernement $\Omega$ en $\Theta$, il s'agit de redéfinir sur $\Omega$ la distribution de masses de croyance initialement définie sur $\Omega$. Pour cela, on utilise l'*extension vide* [Sha76, Sme93] qui est un processus conservatif de réallocation des masses de croyance respectant le PMI. L'opérateur associé est noté « $\uparrow$ » et il est défini par :

$$\begin{array}{rcl} 2^{\Omega} & \rightarrow & 2^{\Theta} \\ m^{\Omega}(B) & \mapsto & m^{\Theta}(\rho(B)) = m^{\Omega\uparrow\Theta}(\rho(B)) \end{array} \tag{2.35}$$

**Exemple 8.** *Dans l'exemple de la figure 2.4, la masse $m^{\Omega}(\omega_2)$ est transférée sur $\{\theta_1, \theta_2\}$ après raffinement. Ce transfère ne privilégie ni $\theta_1$ ni $\theta_2$ justifiant ainsi le terme d'extension* vide *(sous-entendu, l'a priori sur $\{\theta_1, \theta_2\}$ est vide). La masse $m^{\Omega}(\{\omega_1, \omega_2\})$ est quant à elle transférée sur $\rho(\{\omega_1\}) \cup \rho(\{\omega_2\}) = \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5\}$. Après raffinement, la BBA $m^{\Theta}$ est définie par :*

$$\begin{array}{rclrcl} m^{\Theta}(\{\theta_1, \theta_2\}) & = & m^{\Omega}(\{\omega_2\}) & m^{\Theta}(\{\theta_3, \theta_4, \theta_5\}) & = & m^{\Omega}(\{\omega_1\}) \\ & & m^{\Theta}(\{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5\}) = m^{\Omega}(\{\omega_1, \omega_2\}) & & & \end{array} \tag{2.36}$$

**Première utilisation pratique : définition de BBA sur un espace produit** L'opération d'extension vide permet de combiner deux BBA $m_1^{\Omega}$ et $m_2^{\Theta}$ définies sur des cadres de discernement disjoints. Pour cela, les BBA doivent être en premier lieu étendues sur un espace commun défini par *l'espace produit* $\Omega \times \Theta = \{(\omega_k, \theta_i), \forall k \in \{1 \ldots |\Omega|\}, \forall i \{1 \ldots |\Theta|\}\}$. Par exemple, $m_1^{\Omega}$ est redéfinie sur $m_1^{\Omega\times\Theta}$ de la manière suivante :

$$m_1^{\Omega\uparrow\Omega\times\Theta}(C) = \begin{cases} m_1^{\Omega}(B) & \text{si } C = B \times \Theta \\ 0 & \text{sinon} \end{cases} \tag{2.37}$$

La masse initialement allouée à la proposition $B \subseteq \Omega$ est réallouée à $B \times \Theta$ après extension vide minimisant ainsi tout a priori sur $\Theta$ (aucune hypothèse de $\Theta$ n'est favorisée car la source de croyance fournissant $m_1^{\Omega}$ ne connaît rien de $\Theta$). Notons que la transformation $B \subseteq \Theta \mapsto B \times \Omega$ est appelée *extension cylindrique* de $B$ sur $\Omega \times \Theta$.

La combinaison des deux BBA $m_1^{\Omega}$ et $m_2^{\Theta}$ est alors donnée par :

$$\begin{array}{rcl} m_{1 \textcircled{\cap} 2}^{\Omega\times\Theta}(D) & = & (m_1^{\Omega\uparrow\Omega\times\Theta} \textcircled{\cap} m_2^{\Theta\uparrow\Omega\times\Theta})(D) \\ & = & \begin{cases} m_1^{\Omega}(B) \cdot m_2^{\Theta}(C) & \text{si } D = B \times C, B \subseteq \Omega, C \subseteq \Theta \\ 0 & \text{sinon} \end{cases} \end{array} \tag{2.38}$$

**Exemple 9.** *Nous considérons un exemple de reconnaissance de formes où l'on cherche à reconnaître le type d'un panneau de signalisation. Pour cela, on dispose de deux sources de croyance :*

- *une première source de croyance est capable de distinguer deux types de forme : ronde ($R_f$) et triangulaire ($T_f$). Le cadre de discernement est $\Omega = \{R_f, T_f\}$.*
- *une deuxième source discrimine les couleurs : bleu ($B_c$), vert ($V_c$) et magenta ($M_c$). Le cadre de discernement est $\Theta = \{B_c, V_c, M_c\}$.*

*On cherche à combiner conjonctivement les deux premières BBA définies par :*

$$\begin{array}{rclrcl} m_1^{\Omega}(\{R_f\}) & = & 0.3 & m_2^{\Theta}(\{M_c\}) & = & 0.7 \\ m_1^{\Omega}(\{T_f\}) & = & 0.5 & m_2^{\Theta}(\{B_c, V_c\}) & = & 0.1 \\ m_1^{\Omega}(\{R_f, T_f\}) & = & 0.2 & m_2^{\Theta}(\{B_c, V_c, M_c\}) & = & 0.2 \end{array} \tag{2.39}$$

*Pour cela, on utilise l'équation 2.38 : on calcule l'extension vide des BBA sur l'espace produit* $\Omega \times \Theta = \{(R_f, B_c), (R_f, V_c), (R_f, M_c), (T_f, B_c), (T_f, V_c), (T_f, M_c)\}$ *suivie d'une combinaison CRC. La BBA résultante* $m_{1 \circledcirc 2}^{\Omega\times\Theta}$ *comporte* 9 *éléments focaux :*

$$\begin{aligned}
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(T_f, M_c)\}) &= 0.35 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, M_c)\}) &= 0.21 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, M_c), (T_f, M_c)\}) &= 0.14 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(T_f, B_c), (T_f, V_c), (T_f, M_c)\}) &= 0.1 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, B_c), (R_f, V_c), (R_f, M_c)\}) &= 0.06 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(T_f, B_c), (T_f, V_c)\}) &= 0.05 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, B_c), (R_f, V_c), (R_f, M_c), (T_f, B_c), (T_f, V_c), (T_f, M_c)\}) &= 0.04 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, B_c), (R_f, V_c)\}) &= 0.03 \\
m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(R_f, B_c), (R_f, V_c), (T_f, B_c), (T_f, V_c)\}) &= 0.02
\end{aligned} \tag{2.40}$$

*Par exemple,* $m_{1 \circledcirc 2}^{\Omega\times\Theta}(\{(T_f, M_c)\}) = m_1^{\Omega}(\{T_f\}) \cdot m_2^{\Theta}(\{M_c\}) = 0.7 \cdot 0.5 = 0.35$.

**Deuxième utilisation pratique : fusion de BBA avec cadres de discernement hiérarchisés** Le concept de *hiérarchie* (ou de catégorie) est utilisé dans de nombreuses applications et permet de représenter les connaissances sous la forme d'arbres de décision. Cette représentation est facilement *interprétable* et possède une signification réelle pour un utilisateur ou un expert. Un exemple sur les instruments de musique est donné à l'annexe 1. Un exemple similaire mais concernant les automobiles est proposé par Delmotte et Smets [DS04], l'objet de leur article étant l'identification de cibles aériennes. Dans [Mer06], un système hiérarchique de reconnaissance d'adresses postales est proposé où l'aspect hiérarchique permet, d'une part, de reconnaître soit complétement soit partiellement les adresses et, d'autre part, d'estimer les distributions de masses à partir de matrices de confusion.

#### 2.2.2.2 Grossissement et marginalisation

Si $\Theta$ est un raffinement de $\Omega$ on dit que $\Omega$ est un *grossissement* de $\Theta$ (Fig. 2.4). Le grossissement est donc l'opération duale du raffinement et permet de réduire le cadre de discernement en le rendant plus grossier. L'application $\rho$ définie dans le cadre du raffinement n'est pas toujours surjective et par conséquent il existe des éléments $\theta \subseteq \Theta$ qui ne sont pas l'image d'éléments de $\Omega$. Par exemple, sur la figure 2.4, l'élément $\{\theta_1, \theta_3\}$ n'est pas une image par $\rho$ des éléments de $\Omega$. Il existe deux méthodes pour palier ce problème appelées *réductions intérieures* et *extérieures* [Sha76]. Nous encourageons le lecteur intéressé par ces mécanismes à consulter les références [DY02, Quo06] comprenant des exemples détaillés.

L'opération de *marginalisation* est un cas particulier de grossissement lorsque le cadre de discernement avant marginalisation est défini par un espace produit $\Theta \times \Omega$ et que le sous-ensemble d'hypothèses considéré pour le grossissement est un des cadres de l'espace produit. Par exemple, la marginalisation sur $\Omega$ d'une BBA initialement définie sur l'espace produit $\Theta \times \Omega$ est donnée par :

$$m^{\Theta\times\Omega\downarrow\Omega}(B) = \sum_{\substack{C \subseteq \Theta\times\Omega \\ \text{Proj}(C\downarrow\Omega)=B}} m^{\Theta\times\Omega}(C), \quad \forall B \subseteq \Omega \tag{2.41}$$

où $\text{Proj}(C \downarrow \Omega)$ est la projection de $C$ sur $\Omega$. La masse d'un élément $m^{\Omega}(B)$ est obtenue par la somme des masses des éléments de l'espace produit contribuant à $B$.

Il n'est généralement pas possible de retrouver la BBA initiale après marginalisation. Si l'on est intéressé par recouvrir une BBA, le Principe du Minimum d'Information (PMI) impose de choisir la BBA la moins informative de toutes les BBA possibles et cette BBA est donnée par l'extension vide (Eq. 2.38).

**Exemple 10.** *La marginalisation de la BBA $m^{\Omega\times\Theta}_{1\textcircled{\cap}2}$ obtenue dans l'exemple 9 sur $\Omega$ redonne $m^{\Omega}_{1}$ et la marginalisation sur $\Theta$ redonne $m^{\Theta}_{2}$ car les deux BBA ont été étendues avec l'opérateur d'extension vide sur $\Omega\times\Theta$. Par contre, si l'on combine cette BBA avec une BBA définie par $m^{\Omega\times\Theta}_{3}(\{T_f\}\times\Theta)=1$ alors la BBA obtenue est notée $m^{\Omega\times\Theta}_{123}$ (par souci de simplicité, on ne note pas les combinaisons CRC en indices lorsque plus que 2 BBA sont combinées, c'est à dire que l'on assimile $m^{\Omega\times\Theta}_{1\textcircled{\cap}2\textcircled{\cap}3}$ à $m^{\Omega\times\Theta}_{123}$). La BBA $m^{\Omega\times\Theta}_{123}$ est donnée par :*

$$\begin{array}{rclrcl} m^{\Omega\times\Theta}_{123}(\{(T_f,M_c)\}) & = & 0.49 & m^{\Omega\times\Theta}_{123}(\emptyset) & = & 0.3 \\ m^{\Omega\times\Theta}_{123}(\{(T_f,B_c),(T_f,V_c),(T_f,M_c)\}) & = & 0.14 & m^{\Omega\times\Theta}_{123}(\{(T_f,B_c),(T_f,V_c)\}) & = & 0.07 \end{array}$$

*La marginale sur $\Theta$ est :* (2.42)

$$\begin{array}{rclrcl} m^{\Omega\times\Theta\downarrow\Theta}_{123}(\{M_c\}) & = & 0.49 & m^{\Omega\times\Theta\downarrow\Theta}_{123}(\emptyset) & = & 0.3 \\ m^{\Omega\times\Theta\downarrow\Theta}_{123}(\{B_c,V_c,M_c\}) & = & 0.14 & m^{\Omega\times\Theta\downarrow\Theta}_{123}(\{B_c,V_c\}) & = & 0.07 \end{array}$$

*L'extension vide de ces marginales sur $\Omega\times\Theta$ ne redonne pas $m^{\Omega\times\Theta}_{123}$.*

### 2.2.3 Déconditionnement

Disposant d'une BBA conditionnée $m^{\Omega}[B], B\subseteq\Omega$, il n'est généralement pas possible de retrouver la BBA d'origine $m^{\Omega}$ avant conditionnement. Une solution consiste à choisir la BBA la moins informative parmi toutes les BBA possibles et elle est obtenue par l'opérateur de *déconditionnement* [Sme93], noté $\Uparrow$, menant à :

$$m^{\Omega}[B]^{\Uparrow\Omega}(C\cup\overline{B}) = m^{\Omega}[B](C), \quad \forall C\subseteq B \tag{2.43}$$

où $\overline{B}$ est le complémentaire de $B$. La masse initialement affectée à un sous-ensemble $C$ avant déconditionnement (i.e. $m^{\Omega}[B](C)$) est *transférée* sur les sous-ensembles composés de $C$ et ne contenant pas $B$ (i.e. $C\cup\overline{B}$). Cette transformation assure que le (re)conditionnement sur $B$ de la BBA obtenue donne de nouveau $m^{\Omega}[B]$, propriété obtenue grâce au PMI.

Il existe plusieurs applications du déconditionnement. Les plus intéressantes que nous décrivons dans les paragraphes suivants sont :

- *Traduire des règles implicatives en BBA :* comment traduire des implications logiques pondérées en fonctions de croyance ?
- *Fusionner des BBA conditionnelles sur un espace produit :* disposant de plusieurs BBA $m^{\Omega}[\theta_i]$ définies sur un cadre $\Omega$ conditionnellement à des hypothèses $\theta_i$ d'un second cadre de discernement $\Theta$, comment combiner ces BBA ?

Cet opérateur permet par ailleurs d'ajouter des hypothèses dans un cadre de discernement [DS04] : disposant d'une BBA définie $\Omega_0=\{\omega_1,\omega_2\}$, le déconditionnement redéfinit cette BBA sur $\Omega_1=\Omega_0\cup\{\omega_3\}=\{\omega_1,\omega_2,\omega_3\}$. La solution est décrite à l'annexe 2 avec un exemple d'application sur la fusion de BBA dont les hypothèses sont hiérarchisées.

#### 2.2.3.1 Le déconditionnement pour la traduction de règles implicatives

Le déconditionnement a été exploité par Ristic et Smets [RS05b] pour la traduction de règles d'implication dans le cadre de l'identification de cibles. Ce type de règle est très bien adapté pour ajouter des connaissances a priori fournies par des experts.

Une règle d'implication $R$ mettant en jeu deux cadres de discernement $\Omega$ et $\Theta$ est du type :

$$\text{« } R \ : \ \textit{si } \theta \subseteq \Theta \textit{ alors } \omega \subseteq \Omega \textit{ avec une croyance } \alpha \in [0,1] \text{ »}$$

que l'on peut par ailleurs noter par $\theta \Rightarrow \omega$.

A partir de l'expression logique de la règle $R$, il est possible de construire une BBA que l'on pourra ensuite combiner avec d'autres BBA si l'application le nécessite. L'expression de la BBA équivalente à la règle est une BBA conditionnelle :

$$m_R^{\Omega}[\theta](\omega) = \alpha \tag{2.44a}$$

$$m_R^{\Omega}[\theta](\Omega) = 1 - \alpha \tag{2.44b}$$

Dans les systèmes réels, les BBA peuvent être exprimées sur des espaces produits. Si l'on souhaite combiner les BBA conditionnelles de règles implicatives avec d'autres BBA, il peut s'avérer utile d'exprimer les BBA des règles sur l'espace produit $\Omega \times \Theta$. La transformation d'une BBA conditionnelle en une BBA sur un espace produit est réalisée par l'opérateur de déconditionnement [RS05b] qui, appliqué à la règle $R$ (Eq. 2.44), fournit une BBA exprimée sur $\Omega \times \Theta$ et définie par :

$$m_R^{\Omega}[\theta]^{\Uparrow \Omega \times \Theta}(D) = \begin{cases} \alpha & \text{si } \ D = (\omega \times \theta) \cup (\overline{\theta} \times \Omega) \\ 1 - \alpha & \text{si } \ D = (\Omega \times \Theta) \end{cases} \tag{2.45}$$

Pour déterminer et interpréter l'élément focal $D = (\omega \times \theta) \cup (\overline{\theta} \times \Omega)$, nous pouvons écrire la règle $R$ comme l'union de deux règles :

- une première indiquant que si $\theta$ apparaît alors $\omega$ apparaît avec une croyance $\alpha$ : « $R_1$ : *si* $\theta \subseteq \Theta$ *alors* $\omega \subseteq \Omega$ *avec une croyance* $\alpha_1 \in [0,1]$ »,
- une seconde indiquant que si autre chose que $\theta$ apparaît, i.e. $\overline{\theta}$, alors on ne sait rien de la conclusion sur $\Theta$ : « $R_2$ : *si* $\overline{\theta} \subseteq \Theta$ *alors* $\Omega$ *avec une croyance* $\alpha_2 = 1$ ».

Après déconditionnement, la masse $\alpha$ est donc transférée sur l'ensemble $D = (\omega \times \theta) \cup (\overline{\theta} \times \Omega)$ composé de l'ensemble $\omega \times \theta$ pour lequel on dispose d'une connaissance (d'après l'équation 2.44) et de l'ensemble $\overline{\theta} \times \Omega$ représentant l'extension cylindrique de $\overline{\theta}$ sur $\Omega \times \Theta$ et modélisant l'ignorance sur $\Omega$ lorsque $\overline{\theta}$ est vrai. Notons que la BBA issue du déconditionnement respecte le Principe du Minimum d'Information et son conditionnement sur $\theta$ (avec marginalisation sur $\Theta$) redonne la BBA initiale conditionnée [Sme93].

**Exemple 11.** *Nous reconsidérons l'exemple 9 sur la reconnaissance de panneaux de signalisation. En plus des informations sur la* forme *et la* couleur*, on dispose d'une troisième source sous la forme d'une règle implicative spécifiée par un expert. Cette règle indique que* si le panneau est rond alors il est bleu avec une croyance de $0.8$*, les* $0.2$ *manquant représentent l'*ignorance *de l'expert concernant la couleur du panneau sous l'hypothèse que ce dernier soit rond. La règle est donc de la forme :*

$$R \ : \ R_f \ \Rightarrow \ B_c, \ \alpha = 0.8$$

*On souhaite combiner la règle avec la BBA $m^{\Omega\times\Theta}_{1\circledcirc 2}$ donnée à l'équation 2.40 (exemple 9). Pour cela, on procède en trois étapes :*

1. *L'expression de la règle sous la forme d'une BBA conditionnelle :*

$$m^{\Theta}_{R}[R_f](\{B_c\}) = 0.8 \qquad m^{\Theta}_{R}[R_f](\Theta) = 0.2 \tag{2.46}$$

2. *La transformation de cette BBA sur l'espace produit $\Omega \times \Theta$ afin de la combiner avec $m^{\Omega\times\Theta}_{1\circledcirc 2}$. Pour cela, on utilise le déconditionnement de la règle :*

$$m^{\Omega\times\Theta}_{R} = m^{\Theta}_{R}[R_f]^{\Uparrow\Omega\times\Theta} \tag{2.47}$$

*Le résultat du déconditionnement est la BBA suivante :*

$$m^{\Omega\times\Theta}_{R}(\{(R_f,B_c),(T_f,B_c),(T_f,V_c),(T_f,M_c)\}) = 0.8 \qquad m^{\Omega\times\Theta}_{R}(\Omega\times\Theta) = 0.2 \tag{2.48}$$

*où $\Omega \times \Theta = \{(T_f,B_c),(T_f,V_c),(T_f,M_c),(R_f,B_c),(R_f,V_c),(R_f,M_c)\}$ est un espace produit. Cette BBA est équivalente à la forme conditionnelle car son conditionnement par $R_f$ suivie d'une marginalisation sur $\Theta$ redonne la BBA conditionnelle.*

3. *La combinaison par la CRC de la BBA $m^{\Omega\times\Theta}_{1\circledcirc 2}$ avec la règle est exprimée de la façon suivante :*

$$m^{\Omega\times\Theta}_{12R} = m^{\Theta}_{R}[R_f]^{\Uparrow\Omega\times\Theta} \circledcap m^{\Omega\times\Theta}_{1\circledcirc 2} \tag{2.49}$$

*La BBA résultante comporte 13 éléments focaux dont les 5 plus élevés sont les suivants :*

$$\begin{aligned}
m^{\Omega\times\Theta}_{12R}(\{(T_f,M_c)\}) &= 0.462\\
m^{\Omega\times\Theta}_{12R}(\emptyset) &= 0.168\\
m^{\Omega\times\Theta}_{12R}(\{(T_f,B_c),(T_f,V_c),(T_f,M_c)\}) &= 0.1\\
m^{\Omega\times\Theta}_{12R}(\{(R_f,B_c)\}) &= 0.072\\
m^{\Omega\times\Theta}_{12R}(\{(T_f,B_c),(T_f,V_c)\}) &= 0.05\\
&\ldots
\end{aligned} \tag{2.50}$$

*Par exemple,* $m^{\Omega\times\Theta}_{12R}(\{(T_f,M_c)\}) = m^{\Omega\times\Theta}_{1\circledcirc 2}(\{(T_f,M_c)\}) \cdot \Big(m^{\Omega\times\Theta}_{R}(\{(R_f,B_c),(T_f,B_c),(T_f,V_c),(T_f,M_c)\}) + m^{\Omega\times\Theta}_{R}(\Omega\times\Theta)\Big) + m^{\Omega\times\Theta}_{1\circledcirc 2}(\{(R_f,M_c),(T_f,M_c)\}) \cdot m^{\Omega\times\Theta}_{R}(\{(R_f,B_c),(T_f,B_c),(T_f,V_c),(T_f,M_c)\}) = 0.35 + 0.14 \cdot 0.8 = 0.462$.

*Les marginales $m^{\Omega\times\Theta\downarrow\Omega}_{12R}$ et $m^{\Omega\times\Theta\downarrow\Theta}_{12R}$ sur $\Omega$ et $\Theta$ sont données par les BBA suivantes :*

$$\begin{aligned}
m^{\Omega}_{12R}(\{T_f\}) &= 0.612 & m^{\Theta}_{12R}(\{M_c\}) &= 0.532\\
m^{\Omega}_{12R}(\emptyset) &= 0.168 & m^{\Theta}_{12R}(\emptyset) &= 0.168\\
m^{\Omega}_{12R}(\{R_f\}) &= 0.132 & m^{\Theta}_{12R}(\{B_c,V_c,M_c\}) &= 0.152\\
m^{\Omega}_{12R}(\{R_f,T_f\}) &= 0.088 & m^{\Theta}_{12R}(\{B_c,V_c\}) &= 0.076\\
& & m^{\Theta}_{12R}(\{B_c\}) &= 0.072
\end{aligned} \tag{2.51}$$

#### 2.2.3.2 Le déconditionnement pour la fusion de BBA conditionnelles

On suppose disposer d'un jeu de BBA $m^{\Omega}[\theta_i], \forall\theta_i \in \Theta$ définies sur un cadre de discernement $\Omega = \{\omega_1,\omega_2\ldots\}$ conditionnellement à chacun des *singletons* $\theta_i$ d'un deuxième cadre de discernement $\Theta = \{\theta_1,\theta_2\ldots\}$.

Les BBA conditionnelles peuvent être combinées sur l'espace produit $\Omega \times \Theta$ obtenue par la combinaison conjonctive des BBA déconditionnées :

$$m^{\Omega\times\Theta} = \bigcirc\!\!\!\!\bigcap_i m^{\Omega}[\theta_i]^{\Uparrow\Omega\times\Theta} \tag{2.52}$$

Chaque sous-ensemble $S$ de l'espace produit $\Omega \times \Theta$ se voit allouer la masse :

$$m^{\Omega\times\Theta}(S) \;=\; \prod_{\theta_i\in\Theta} m^{\Omega}[\theta_i](v), \quad \forall S \subseteq \Omega \times \Theta \tag{2.53}$$

où $v = ((\theta_i \times \Omega) \cap S)^{\downarrow\Omega}$. Cette expression analytique des éléments de la BBA résultante a été dérivée par Smets [Sme78, Sme93]. De la même manière, l'expression analytique de la communalité des sous-ensemble $S \subseteq \Omega \times \Theta$ est donnée par :

$$q^{\Omega\times\Theta}(S) \;=\; \prod_{\theta_i\in\Theta} q^{\Omega}[\theta_i](v), \quad \forall S \subseteq \Omega \times \Theta \tag{2.54}$$

L'opération de déconditionnement utilisée de cette manière a permis de dériver le Théorème de Bayes Généralisé (§ 2.2.1) [Sme93]. Notons que les croyances conditionnelles sont supposées distinctes. Si ce n'est pas le cas, il faut utiliser la *combinaison conjonctive prudente* (Eq. 2.64).

**Exemple 12.** *Nous reprenons l'exemple de la prédiction du temps décrit dans le paragraphe 2.2.1.1 et notamment les croyances conditionnelles $m^{\Omega_j}[E_{j-1}]$, $m^{\Omega_j}[P_{j-1}]$ et $m^{\Omega_j}[N_{j-1}]$ données dans le tableau 2.4. Il est possible de déterminer la BBA équivalente sur $\Omega_{j-1} \times \Omega_j$ à l'ensemble de ces BBA conditionnelles. Pour cela, deux étapes sont nécessaires : déconditionner chaque BBA puis les combiner. On a donc :*

$$m^{\Omega_{j-1}\times\Omega_j} \;=\; m^{\Omega_j}[E_{j-1}]^{\Uparrow\Omega_{j-1}\times\Omega_j} \bigcirc\!\!\!\!\bigcap m^{\Omega_j}[P_{j-1}]^{\Uparrow\Omega_{j-1}\times\Omega_j} \bigcirc\!\!\!\!\bigcap m^{\Omega_j}[N_{j-1}]^{\Uparrow\Omega_{j-1}\times\Omega_j} \tag{2.55}$$

*Le résultat de la combinaison comporte environ* 343 *éléments focaux (potentiellement, il existe* $2^9$ *éléments) et nous présentons seulement les* 6 *plus grandes valeurs approchées ci-dessous :*

$$\begin{aligned}
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},N_j),(P_{j-1},N_j),(E_{j-1},E_j)\}) &= 0.168\\
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},N_j),(P_{j-1},P_j),(E_{j-1},E_j)\}) &= 0.116\\
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},N_j),(P_{j-1},N_j),(E_{j-1},N_j),(E_{j-1},E_j)\}) &= 0.060\\
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},N_j),(P_{j-1},P_j),(E_{j-1},N_j),(E_{j-1},E_j)\}) &= 0.041\\
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},N_j),(P_{j-1},N_j),(P_{j-1},E_j),(E_{j-1},E_j)\}) &= 0.041\\
m^{\Omega_{j-1}\times\Omega_j}(\{(N_{j-1},E_j),(P_{j-1},N_j),(E_{j-1},E_j)\}) &= 0.037
\end{aligned} \tag{2.56}$$

Dans la suite du manuscrit, nous exploiterons les concepts définis dans cette section et en particulier ceux qui concernent le conditionnement/déconditionnement jetant les premières bases du filtre temporel crédibiliste (chapitre 3), de l'ordonnanceur d'états crédibilistes (chapitre 4) et des modèles de Markov cachés adaptés aux fonctions de croyance (chapitre 5).

### 2.2.4 Le PMI pour transformer une distribution de probabilités en BBA

L'estimation des distributions de masses (BBA) est un problème difficile. Une solution consiste à exploiter les outils performants d'estimation de distributions de probabilités puis de transformer ces distributions en BBA. La transformation « probabilités → BBA » est nécessaire car les combinaisons

CRC et DRC perdent leur intérêt lorsque les BBA combinées sont Bayésiennes.

Disposant d'une distribution de probabilités pignistiques $\mathbf{BetP}^{\Omega}$, le problème est de déterminer une BBA $m^{\Omega}$ dont la transformation pignistique redonne $\mathbf{BetP}^{\Omega}$. Il existe généralement plusieurs BBA qui vérifient cette propriété et, dans cette *famille*, le Principe du Minimum d'Information (PMI) impose de choisir la BBA qui est la moins *engagée*, c'est à dire la moins informative.

L'expression de la BBA est définie formellement de la façon suivante (théorème 7.1 de [Sme00c, Sme00b]) : disposant d'une distribution de probabilités pignistiques $\mathbf{BetP}^{\Omega}$ sur les singletons $\omega_1, \omega_2 \ldots \omega_K$ de $\Omega$ (avec $K = |\Omega|$) dont les éléments de la distribution sont ordonnés :

$$\mathbf{BetP}^{\Omega}(\omega_1) \geq \mathbf{BetP}^{\Omega}(\omega_2) \geq \cdots \geq \mathbf{BetP}^{\Omega}(\omega_k) \geq \cdots \geq \mathbf{BetP}^{\Omega}(\omega_K)$$

Le PMI permet de construire la BBA la moins engagée $\hat{m}^{\Omega}$ associée à $\mathbf{BetP}^{\Omega}$ composée d'éléments focaux emboîtés $A = \{\omega_1, \omega_2 \ldots \omega_k\}$ tels que :

$$\hat{m}^{\Omega}(A) = |A| \cdot \Big(\mathbf{BetP}^{\Omega}(\omega_k) - \mathbf{BetP}^{\Omega}(\omega_{k+1})\Big) \tag{2.57}$$

où $\mathbf{BetP}^{\Omega}(\omega_{K+1}) = 0$ par définition. Smets [Sme00b, Sme00c] a montré que cette BBA est *consonante* et la distribution de plausibilités associée est une distribution de possibilités. Notons qu'une méthode a été proposée pour passer d'une distribution de possibilités à une distribution de masses dont l'expression est assez similaire [DP82] à l'expression précédente.

**Exemple 13.** *On pose $\Omega = \{\omega_1, \omega_2, \omega_3\}$ et on définit la distribution de probabilités pignistiques ainsi que les deux BBA suivantes :*

$$\begin{array}{rclrclrcl}
\mathbf{BetP}^{\Omega}(\{\omega_1\}) & = & 0.6 & m_1^{\Omega}(\{\omega_1\}) & = & 0.3 & m_2^{\Omega}(\{\omega_1\}) & = & 0.45 \\
\mathbf{BetP}^{\Omega}(\{\omega_2\}) & = & 0.2 & m_1^{\Omega}(\{\omega_1, \omega_2\}) & = & 0.4 & m_2^{\Omega}(\{\omega_1, \omega_2\}) & = & 0.1 \\
\mathbf{BetP}^{\Omega}(\{\omega_3\}) & = & 0.2 & m_1^{\Omega}(\{\omega_1, \omega_3\}) & = & 0.2 & m_2^{\Omega}(\{\omega_2, \omega_3\}) & = & 0.2 \\
 & & & m_1^{\Omega}(\{\omega_3\}) & = & 0.1 & m_2^{\Omega}(\{\omega_1, \omega_3\}) & = & 0.1 \\
 & & & & & & m_2^{\Omega}(\Omega) & = & 0.15
\end{array}$$

*Les BBA $m_1^{\Omega}$ et $m_2^{\Omega}$ ont la même distribution de probabilités pignistiques définie par $\mathbf{BetP}^{\Omega}$.*

*Afin de faire un choix entre les deux BBA, on calcule la non-spécificité (Eq. 2.8). La non-spécificité de $m_1^{\Omega}$ est*

$$N(m_1) = (0.3 + 0.1) \cdot \log_2(1) + (0.4 + 0.2) \cdot \log_2(2) = 0.6$$

*et celle de $m_2^{\Omega}$ est*

$$N(m_2) = 0.45 \cdot \log_2(1) + (0.1 + 0.2 + 0.1) \cdot \log_2(2) + 0.15 \cdot \log_2(3) = 0.6377$$

*Par conséquent, la BBA la moins engagée est $m_2$ (puisqu'elle est la moins spécifique) et correspond à celle que l'on devrait choisir si on avait seulement ces deux BBA dans la famille des BBA possibles. En reprenant l'expression 2.57, on détermine la BBA la moins informative parmi toutes les BBA possibles qui est donnée par la BBA consonante définie par :*

$$m_3^{\Omega}(\{\omega_1\}) = 0.4 \qquad m_3^{\Omega}(\Omega) = 0.6 \tag{2.58}$$

*dont la non-spécificité est*

$$N(m_3) = 0.951 > N(m_2) > N(m_1)$$

*La masse sur $\{\omega_1\}$ est par exemple obtenue de la manière suivante : $m_3^{\Omega}(\{\omega_1\}) = 1 \cdot (0.6 - 0.2) = 0.4$ et celle sur $\Omega$ par $m_3^{\Omega}(\Omega) = 3 \cdot (0.2 - 0) = 0.6$.*

### 2.2.5 Combinaison de sources non distinctes

Un grand nombre de mécanismes décrits dans les paragraphes précédents étaient basés sur des règles de combinaison conjonctive ou disjonctive sous l'hypothèse que les sources de croyance étaient *distinctes*, concept proche de l'*indépendance* en statistique [Sme93, YSM01, YSM02a, YSM02b].

Une règle de combinaison conjonctive dite *prudente* (CCRC : *Cautious Conjunctive Rule of Combination*), notée $\textcircled{\wedge}$, et une règle de combinaison disjonctive *hardie* (BDRC : *Bold Disjunctive Rule of Combination*), notée $\textcircled{\vee}$, ont été proposées par Denœux [Den07] (à partir de travaux initiés par Smets [Sme95]) pour combiner des sources *non-distinctes*. Ces règles sont associatives, commutatives et surtout *idempotentes* à la différence des règles usuelles CRC et DRC. Les quatre règles et leur utilisation sont représentées dans le tableau 2.9.

| | sources fiables | sources non fiables |
|---|---|---|
| sources distinctes | $\textcircled{\cap}$ | $\textcircled{\cup}$ |
| sources non distinctes | $\textcircled{\wedge}$ | $\textcircled{\vee}$ |

**TAB. 2.9 :** *Les règles de combinaison et leur contexte d'utilisation [Den07].*

Les règles CCRC et BDRC sont basées sur la décomposition canonique des BBA et sur des opérateurs flous que nous décrivons dans les paragraphes suivants.

#### 2.2.5.1 Décomposition canonique d'une BBA

L'idée de la décomposition canonique [Sha76] est d'exprimer une BBA non-dogmatique ($m^{\Omega}(\Omega) > 0$) sous la forme d'une combinaison conjonctive ou disjonctive de BBA simples $B^w$. On rappelle que $B^w$ est une écriture simplifiée signifiant que la BBA est formée de deux éléments focaux : $m^{\Omega}(B) = 1 - w$ et $m^{\Omega}(\Omega) = w$.

Pour définir la décomposition canonique de BBA quelconques, Smets [Sme95] a proposé le concept de fonction simple généralisée (GSBBA : *generalized simple BBA*) définie par une fonction $\mu : 2^{\Omega} \rightarrow \Re$ et vérifiant les contraintes suivantes :

$$\mu(B) = 1 - w \tag{2.59a}$$

$$\mu(\Omega) = w \tag{2.59b}$$

$$\mu(C) = 0, \quad \forall C \in 2^{\Omega} \backslash \{B, \Omega\} \tag{2.59c}$$

avec $B \neq \Omega$ et $w \in [0, +\infty[$. Lorsque $w \leq 1$, la BBA obtenue est une BBA *simple*.

La *décomposition canonique conjonctive* d'une BBA non-dogmatique, qui existe et qui est unique, est définie comme la combinaison conjonctive de GSBBA :

$$m^{\Omega} = \bigcirc\!\!\!\!\bigcap_{B \subset \Omega} B^{w(B)} \tag{2.60}$$

où les *poids* de la décomposition canonique conjonctive $w(B) \in [0, +\infty), \forall B \subset \Omega$ sont définis par l'équation suivante :

$$w^{\Omega}(B) = \prod_{B \subseteq C} q^{\Omega}(C)^{(-1)^{|C|-|B|+1}}, \quad \forall B \subseteq \Omega \tag{2.61}$$

De la même manière, la *décomposition canonique disjonctive* [Den07] d'une BBA *sous normale* ($m^\Omega(\emptyset) > 0$) existe, est unique et elle est définie par :

$$m^\Omega = \bigcirc\!\!\!\!\bigcup_{B \neq \emptyset} B_{v(B)} \tag{2.62}$$

où la notation $B_{v(B)}$ représente une distribution de masses avec deux éléments focaux : $m^\Omega(\emptyset) = v(B)$ et $m^\Omega(B) = 1 - v(B)$, et les poids $v(B)$ sont donnés par :

$$v(B) = \overline{w}(\overline{B}) \tag{2.63}$$

Pour calculer les poids $v(B)$ il suffit de calculer la BBA $\overline{m}^\Omega$, puis les poids associés $\overline{w}(B)$ par l'équation 2.61 et enfin de prendre le complémentaire de chaque poids (voir [Den07] pour des exemples, cf. exemple 14 ci-après).

#### 2.2.5.2 Définitions des règles prudente (CCRC) et hardie (BDRC)

La CCRC et la BDRC sont basés sur des opérateurs flous de type *t-norme* (opérateur conjonctif flou) et *t-conorme* (opérateur disjonctif flou). Denœux [Den07] a ainsi montré qu'une infinité d'opérateurs de combinaison conjonctive « prudente » peut être déterminée conférant au TBM un large ensemble de méthodes de combinaison de fonctions de croyance. Dans ce paragraphe, nous nous limitons à un cas particulier : celui de la t-norme « minimum » utilisée à la fois dans la CCRC et la BDRC.

La CCRC $\circledwedge$ est définie de la manière suivante :

$$m_1^\Omega \circledwedge m_2^\Omega = \bigcirc\!\!\!\!\bigcap_{B \subset \Omega} B^{w_1^\Omega(B) \wedge w_2^\Omega(B)} \tag{2.64}$$

où $\wedge$ est l'opérateur « minimum ». Notons que la BBA vide n'est pas l'élément neutre de la CCRC [Den07].

La BDRC $\circledvee$ est donnée par :

$$m_1^\Omega \circledvee m_2^\Omega = \bigcirc\!\!\!\!\bigcup_{B \neq \emptyset} B_{v_1(B) \wedge v_2(B)} \tag{2.65}$$

où les poids $v_i(B)$ sont calculés à partir des poids $w_i(B)$ par l'équation 2.63.

**Exemple 14.** *Nous reprenons l'exemple de la prédiction du temps. On dispose de deux sources d'information $m_1^{\Omega_j}$ et $m_2^{\Omega_j}$ sur le temps de la journée définies sur $\Omega_j = \{E_j, P_j, N_j\}$ et que l'on souhaite combiner pour améliorer la prise de décision sur le temps au jour $j$.*

*Le tableau ci-dessous donne les détails sur les différentes fonctions utilisées dans la CCRC et la BDRC et notamment les BBA $m_i$, leur négation $\overline{m}_i$, et les poids des décompositions ($w_i$ et $v_i$). On remarque que certains poids ($w_i$ et $v_i$) sont supérieurs à l'unité ce qui signifie que la BBA associée est une GSBBA.*

| | $m_1^\Omega$ | $m_2^\Omega$ | $\overline{m}_1^\Omega$ | $\overline{m}_2^\Omega$ | $w_1^\Omega$ | $w_2^\Omega$ | $\overline{w}_1^\Omega$ | $\overline{w}_2^\Omega$ | $v_1^\Omega$ | $v_2^\Omega$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $\emptyset_j$ | 0.07 | 0.01 | 0.15 | 0.46 | 0.93 | 1 | 2.61 | 6.66 | 1 | 1 |
| $\{E_j\}$ | 0 | 0 | 0.56 | 0 | 1 | 1.134 | 0.111 | 1 | 1 | 1 |
| $\{P_j\}$ | 0 | 0 | 0 | 0.36 | 1.883 | 1 | 1 | 0.027 | 1 | 1 |
| $\{E_j, P_j\}$ | 0.22 | 0.17 | 0 | 0 | 0.405 | 0.73 | 1 | 1 | 0.241 | 0.056 |
| $\{N_j\}$ | 0 | 0 | 0.22 | 0.17 | 1 | 1 | 0.241 | 0.056 | 1 | 1 |
| $\{E_j, N_j\}$ | 0 | 0.36 | 0 | 0 | 1 | 0.561 | 1 | 1 | 1 | 0.027 |
| $\{P_j, N_j\}$ | 0.56 | 0 | 0 | 0 | 0.211 | 1 | 1 | 1 | 0.111 | 1 |
| $\Omega_j$ | 0.15 | 0.46 | 0.07 | 0.01 | 1 | 1 | 1 | 1 | 2.61 | 6.66 |

*Les résultats de la combinaison conjonctive par la règle conjonctive usuelle (CRC) et la règle prudente (CCRC) sont donnés dans le tableau de gauche ci-dessous. Nous donnons aussi les résultats de la combinaison disjonctive usuelle (DRC) et hardie (BDRC). Pour le cas conjonctif, nous présentons les pignistiques dans le tableau de droite.*

| | $m^\Omega_{1 \textcircled{\scriptsize o} 2}$ | $m^\Omega_{1 \textcircled{\scriptsize \wedge} 2}$ | $m^\Omega_{1 \textcircled{\scriptsize \cup} 2}$ | $m^\Omega_{1 \textcircled{\scriptsize \vee} 2}$ |
|---|---|---|---|---|
| $\emptyset$ | 0.079 | 0.261 | 0.001 | 0 |
| $\{E_j\}$ | 0.079 | 0.051 | 0 | 0 |
| $\{P_j\}$ | 0.095 | 0.245 | 0 | 0 |
| $\{E_j, P_j\}$ | 0.164 | 0.066 | 0.051 | 0.007 |
| $\{N_j\}$ | 0.202 | 0.131 | 0 | 0 |
| $\{E_j, N_j\}$ | 0.054 | 0.035 | 0.025 | 0.016 |
| $\{P_j, N_j\}$ | 0.258 | 0.167 | 0.006 | 0.003 |
| $\Omega_j$ | 0.069 | 0.045 | 0.917 | 0.973 |

| | $\mathbf{BetP}_1$ | $\mathbf{BetP}_2$ | $\mathbf{BetP}_{1 \textcircled{\scriptsize o} 2}$ | $\mathbf{BetP}_{1 \textcircled{\scriptsize \wedge} 2}$ |
|---|---|---|---|---|
| $\{E_j\}$ | 0.17 | **0.42** | 0.23 | 0.16 |
| $\{P_j\}$ | **0.47** | 0.24 | 0.36 | **0.51** |
| $\{N_j\}$ | 0.36 | 0.34 | **0.41** | 0.33 |

*Les transformations pignistiques des BBA combinées ($m_1^\Omega$ et $m_2^\Omega$) et des BBA obtenues par les combinaisons conjonctives ($m^\Omega_{1 \textcircled{\scriptsize o} 2}$, $m^\Omega_{1 \textcircled{\scriptsize \wedge} 2}$) mènent à une décision différente (voir les valeurs en gras, tableau en bas à droite) :* pluvieux *si la décision est prise sur $m_1$,* ensoleillé *si la décision est prise sur $m_2$,* nuageux *si la décision est prise sur le résultat de la CRC et* pluvieux *si la décision est prise sur le résultat de la CRCC. La décision $P_j$ à partir des pignistiques sur le résultat de la CCRC est en accord avec la décision sur $m_1$ cependant, la valeur du conflit en sortie de la CCRC reste élevée ($0.261$). La CRC favorise quant à elle un élément ($N_j$) sur lequel on aurait pas forcément parié avant fusion (relativement aux pignistiques sur les masses initiales). Si on analyse les deux BBA initiales, on s'aperçoit qu'elles sont plutôt conflictuelles concernant $E_j$ et $P_j$ et plutôt concensuelles concernant $N_j$. Ceci explique que, par la CRC, c'est $N_j$ qui l'emporte (consensus par l'intersection).*

Ces opérateurs s'appliquant aux sources non distinctes sont particulièrement intéressants dans le cadre des extensions de méthodes probabilistes où des hypothèses d'indépendance sont souvent émises pour simplifier les calculs (et pour appliquer l'opérateur « produit »). C'est le cas, par exemple, dans le cadre des réseaux Bayésiens [Mur02] pour lesquels il est presque systématiquement admis que les distributions observées sont indépendantes conditionnellement aux états. Dans le cadre des réseaux évidentiels [XS94, Sme93] (qui étendent les réseaux Bayésiens aux fonctions de croyance), les distributions de croyance non indépendantes peuvent être combinées avec la BDRC ou la CCRC (en fonction des cas). La CCRC et la BDRC représentent alors des alternatives intéressantes de gestion de l'indépendance avec un panel d'opérateurs possibles.

### 2.2.6 Affaiblissement généralisé dit *contextuel*

L'affaiblissement *simple* décrit dans le paragraphe 2.1.4.4 agit de la même manière (avec le même coefficient) quelle que soit l'hypothèse. La généralisation de l'affaiblissement est basée sur l'idée que la fiabilité d'une source peut agir différemment sur les différentes hypothèses du cadre de discernement [MQD07, Mer06].

La généralisation passe par deux étapes : la définition d'une partition notée $\Theta = \{\theta_1, \theta_2, \ldots, \theta_G\}$ du cadre de discernement $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ et la quantification des coefficients d'affaiblissement $\alpha_g, g \in \{1 \ldots G\}$ sur chaque élément de la partition $\Theta$. Chaque élément $\theta_g \in \Theta$ de la partition se voit donc affecter un taux d'affaiblissement $\alpha_g$.

Disposant de $G$ éléments $\theta_g$ dans la partition et donc de $G$ taux d'affaiblissement rassemblés dans le vecteur $\boldsymbol{\alpha} = [\alpha_1 \ \ldots \ \alpha_g \ \ldots \ \alpha_G]$, l'affaiblissement d'une BBA $m^\Omega$ génère une BBA notée ${}^{\boldsymbol{\alpha}}_{\Theta}m^\Omega$ définie par :

$$ {}^{\boldsymbol{\alpha}}_{\Theta}m^\Omega = m^\Omega \textcircled{\cup} m_1^\Omega \textcircled{\cup} m_2^\Omega \ldots \textcircled{\cup} m_g^\Omega \ldots \textcircled{\cup} m_G^\Omega \tag{2.66} $$

où les BBA $m_g^\Omega, g \in \{1 \ldots G\}$ sont définies par :

$$ \begin{array}{rcl} m_g^\Omega(\emptyset) & = & 1 - \alpha_g \\ m_g^\Omega(\theta_g) & = & \alpha_g \end{array} \tag{2.67} $$

Lorsque la partition est égale à $\Omega$, i.e. $\Theta = \{\Omega\}$ (il n'existe donc qu'un seul coefficient de fiabilité), l'affaiblissement généralisé devient un affaiblissement simple (§ 2.1.4.4).

**Exemple 15.** *On considère la BBA $m_2^\Omega$ définie dans l'exemple 14 où $\Omega = \{E_j, P_j, N_j\}$. On considère que la source de croyance fournissant $m_2^\Omega$ n'est fiable qu'à $60\%$ en ce qui concerne l'hypothèse $E_j$ et qu'à $85\%$ pour les hypothèses $P_j$ et $N_j$. On créé alors une partition $\Theta = \{\theta_1, \theta_2\}$ où $\theta_1 = \{E_j\}$ et $\theta_2 = \{P_j, N_j\}$. Les coefficients d'affaiblissement sont $\boldsymbol{\alpha} = [0.4, \ 0.15]$ et la BBA affaiblie ${}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega$ a les éléments focaux suivants :*

$$ \begin{array}{rclrcl} {}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega(\emptyset) & = & 0.0051 & {}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega(\{E_j\}) & = & 0.0034 \\ {}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega(\{E_j, P_j\}) & = & 0.1870 & {}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega(\{P_j, N_j\}) & = & 0.3729 \\ {}^{\boldsymbol{\alpha}}_{\Theta}m_2^\Omega(\Omega) & = & 0.4316 & & & \end{array} \tag{2.68} $$

*La masse sur $E_j$ était nulle avant affaiblissement et devient positive (faiblement) après le processus d'affaiblissement contextuel car ce dernier est une généralisation qui a pour effet de transférer des masses vers des sur-ensembles (la non-spécificité augmente). Ainsi, la masse $0.01$ allouée à l'ensemble vide avant affaiblissement est en partie transférée vers $E_j$ qui se voit allouer une masse de $0.01 \cdot (0.4 \cdot 0.85) = 0.0034$ par le processus d'affaiblissement (Eq. 2.66 suivie de Eq. 2.67).*

## 2.3 Conclusion

Le Modèle des Croyances Transférables (TBM) permet de représenter et combiner la connaissance à l'aide de fonctions de croyance. Le TBM offre des outils pour combiner des BBA définies sur des espaces différents, pouvant être plus ou moins précis ou grossiers. Des opérateurs de combinaison conjonctif et disjonctif permettent par ailleurs de fusionner des BBA provenant de sources pouvant être plus ou moins fiables et non-distinctes. Ces mécanismes de base ont été utilisés pour de

nombreux problèmes (voir l'état de l'art dressé par Vannoorenberghe [Van03] en 2003) et notamment en reconnaissance des formes et classification [Den04, Quo06, DS06].

Dans la suite du manuscrit, nous exploitons ces mécanismes pour développer des méthodes de reconnaissance de séquences dans le cadre de l'analyse de systèmes dynamiques.

# Chapitre 3

# Filtrage de fonctions de croyance

## 3.1 Introduction

Un système physique peut être décrit par un ensemble de variables reflétant l'*état* du système à un instant. Par exemple, une automobile est un système physique dont l'état peut être *à l'arrêt*, *en marche*, *roule*, *freine*, *double un véhicule*, *prend un rond-point* ... Une fonction de *transition* permet par ailleurs de décrire l'*évolution* du système au cours du *temps*. Par exemple, la transition *double un véhicule* → *prend un rond-point* est impossible (on se place dans le cas de conducteur raisonné) alors que la transition *freine* → *à l'arrêt* est probable (possible ou plausible).

Un *système dynamique* est un système physique évoluant au cours du temps. L'analyse d'un tel système a pour objectif de déterminer les connaissances sur l'état courant en prenant en compte les informations disponibles, entre autres les liens sur les connaissances entre deux instants consécutifs et les liens entre les états. Dans le cadre de l'approche probabiliste, ce lien est classiquement modélisé statistiquement par une distribution de probabilités sur les états à un instant conditionnellement aux états à l'instant précédent. L'évolution de la distribution de probabilités sur les états à chaque instant dépend non seulement des probabilités sur les états à l'instant précédent, pondérées par la fonction de transition, mais dépend aussi des *observations* mesurées sur le système à l'aide capteurs. Les observations sont généralement liées aux états à l'aide de *modèles d'observation* générant, pour une observation donnée, une distribution de probabilités sur les états. La problématique du filtrage consiste à améliorer la connaissance sur les états à partir des observations, des transitions entre deux instants consécutifs et des liens entre les états. L'analyse de systèmes dynamiques basée sur les probabilités a été largement traitée [Sch06] comme par exemple avec le filtrage de Kalman ou le filtre particulaire [Sch06, SKG06].

Nous considérons dans ce chapitre que l'ensemble de ces informations (observations, fonctions de transition et distributions sur les états) sont représentées par des *fonctions de croyance* au lieu des probabilités. On cherche donc à déterminer la distribution de croyances sur les états à un instant connaissant les informations passées, courantes et liens entre les états.

Nous considérons les méthodes de filtrage de type Markovien[1] ce qui signifie que le calcul de la BBA (*Basic Belief Assignment*) $m^{\Omega_t}$ en sortie du filtre à $t$ est effectué à partir de :

- La BBA en sortie du filtre à l'instant précédent $m^{\Omega_{t-1}}$,
- La BBA correspondant à la *mesure* courante $\tilde{m}^{\Omega_t}$ et que l'on cherche à filtrer,

[1] L'aspect Markovien permet de limiter les coûts de calcul et de stockage mémoire.

– Un modèle d'évolution liant les BBA $m^{\Omega_{t-1}}$ et $m^{\Omega_t}$ à deux instants consécutifs.

Dans le schéma classique de filtrage, deux étapes sont généralement distinguées :

– *Prédiction :* calcul de la *BBA prédite* $\hat{m}^{\Omega_t}$ à l'aide d'un *modèle d'évolution* de la BBA $m^{\Omega_{t-1}}$ entre $t-1$ et $t$.
– *Mise à jour :* fusion de la BBA prédite $\hat{m}^{\Omega_t}$ avec la *BBA mesurée* $\tilde{m}^{\Omega_t}$ à l'aide d'une combinaison conjonctive.

L'analyse de systèmes dynamiques basée sur les fonctions de croyance n'a été que très peu abordée jusqu'à présent. Nous présentons deux techniques crédibilistes développées récemment par Smets.

**Filtre de Kalman** Le filtre de Kalman (KF : *Kalman filter*) est initialement une méthode probabiliste très répandue pour le filtrage de données continues et souvent appliquée au suivi d'objets (filtrage de leur position et de leur vitesse). Rappelons que le KF initial est basé sur les hypothèses que les caractéristiques observées sont indépendantes conditionnellement aux éléments de $\Omega_t$ et que les différents bruits sont Gaussiens. De plus, des KF plus élaborés ont été proposés pour le cas non linéaire ou non Gaussien dans le cadre des probabilités [Sch06, SKG06].

Le KF a été étendue aux fonctions de croyance[2] par Smets et Ristic [RS04]. Les relations mathématiques des deux formalismes sont très proches à la différence de la phase de classification où le TBM permet de traduire bien plus aisément le comportement des objets suivis grâce au Théorème de Bayes Généralisé (GBT) et aux règles d'implication. La mise à jour consiste à combiner la BBA mesurée $\tilde{m}^{\Omega_t}$ avec la prédiction en utilisant une combinaison conjonctive. Cette combinaison permet de quantifier l'adéquation entre le modèle utilisé pour la prédiction et les données mesurées. Dans le KF, l'erreur entre la prédiction et la mesure est exploitée pour adapter la prédiction à l'instant suivant.

**Mémoire à facteur d'oubli** La méthode de filtrage la plus commune dans le cadre des fonctions de croyance temporelles a été proposée par Smets [Sme05a] et elle est basée sur la notion de *memory decay* (*mémoire à facteur d'oubli*). L'article [Sme05a] décrit des méthodes de résolution du conflit lors de l'application de la règle de combinaison conjonctive (CRC). Le problème observé est le suivant : une source émet une BBA $\tilde{m}^{\Omega_t}$ à des instants $t$ discrets *non régulièrement espacés* et il s'agit de déterminer la probabilité pignistique sur les hypothèses de $\Omega_t$ à chaque instant en prenant en compte les BBA passées et la BBA courante[3]. Smets propose tout d'abord de combiner ces BBA, présentes et passées, conjonctivement avec la CRC. Cependant, la CRC n'est pas idempotente et les CRC successives peuvent générer du conflit qui ne fera qu'augmenter jusqu'à l'unité car cet élément est absorbant par la conjonction. Pour solutionner le problème du conflit, Smets propose d'utiliser une mémoire à facteur d'oubli : la BBA émise à un instant $t = t_1$ n'a plus la même signification qu'à $t = t_2$ où $t_2 \geq t_1 + \Delta_t$ et $\Delta_t > 1$. Cet *oubli* est représenté par un *affaiblissement* : la BBA précédente, par exemple à $t = t_1$, est affaiblie *proportionnellement à l'intervalle de temps* $\Delta_t = t_2 - t_1$ séparant les deux émissions successives d'une BBA par la source. L'affaiblissement a pour rôle de transférer de la masse sur le doute et de diminuer la masse sur l'ensemble vide limitant ainsi le conflit à la prochaine combinaison. La méthode de Smets respecte le schéma « prédiction - mise à jour » car nous pouvons

[2]Les fonctions de croyance utilisées sont continues [Sme05b].

[3]Une extension ce problème est proposée dans [Sme05d] pour des sources défectueuses et bruitées.

interpréter la BBA affaiblie comme une prédiction[4]. Dans cette méthode, le conflit est simplement affaibli mais n'est pas exploité pour le filtrage.

**Notes sur les modèles à sauts probabilistes** Les distributions temporelles reflétant l'état de systèmes dynamiques présentent des périodes de stationnarité. Cette hypothèse est à la base des *modèles à sauts* (SLDS : *switching Linear Dynamical System*) [PRCM99, MB07]. Ce phénomène de stationnarité s'observe sur les figures 3.1a et 3.3a :

- sur l'intervalle $[600, 900]$ de la figure 3.1a et sur l'intervalle $[60, 90]$ de la figure 3.3a, la croyance devrait être globalement allouée à l'hypothèse $F_t$ signifiant « l'état est faux ».
- sur l'intervalle $[300, 600]$ de la figure 3.1a et sur l'intervalle $[50, 60]$ de la figure 3.3a, la croyance devrait être globalement allouée à l'hypothèse $T_t$ signifiant « l'état est vrai ».

Les SLDS représentent une forme de filtre de Kalman où plusieurs modèles d'évolution peuvent être appliqués à un même instant $t$. L'idée est d'utiliser un modèle d'évolution adapté aux données traitées et de changer le modèle au gré de l'évolution de ces données. Ces modèles ont été peu appliqués car la complexité de l'inférence est élevée même pour des données temporelles de faible durée. En effet, le problème de l'inférence est de déterminer quel modèle appliquer à chaque instant or, sous l'hypothèse que deux modèles sont possibles à chaque instant et que la durée du filtrage est $D$, il existe $2^D$ séquences de modèles possibles (à chaque instant, deux modèles sont possibles). De plus, il faut garder à l'esprit que chaque modèle est appliqué sur un ensemble d'états et donc l'inférence doit aussi déterminer quel est le meilleur état avec le modèle en cours. Notons que dans [PRCM99], un algorithme d'inférence approximatif (de type Viterbi) a été proposé pour cette tâche dans le cadre de la théorie des probabilités sous des hypothèses similaires à celles supposées dans le filtre de Kalman.

Dans la section suivante, nous décrivons le principe général du filtre temporel crédibiliste (TCF : *Temporal Credal Filter*) à l'aide d'un exemple. Puis, nous détaillons les mécanismes du filtre et l'estimation de ses paramètres. Enfin, nous proposons quelques pistes d'amélioration.

## 3.2 Filtrage crédibiliste avec modèles à sauts

Comme pour le filtre de Kalman et pour la méthode de Smets, le filtre que nous proposons s'appuie sur le principe « prédiction - mise à jour » mais auquel nous avons adjoint un système de modèle à sauts pour le calcul de la prédiction.

Dans le cadre de l'adaptation des modèles à sauts pour le filtrage des fonctions de croyance, nous proposons de réduire la complexité des mécanismes des SLDS en considérant des commutations de modèles binaires et catégoriques : une commutation est réalisée à partir du *seuillage* d'un critère adapté. Nous n'imposons aucune hypothèse sur la forme de bruit ni sur l'indépendance des observations conditionnellement aux états à la différence des SLDS. Les paramètres que nous utilisons dans le filtre proposé sont peu nombreux et facilement estimables.

La commutation est basée sur le conflit entre prédiction et mesure. Contrairement aux méthodes usuelles de filtrage de fonctions de croyance temporelles, nous exploitons le conflit comme une source d'information pertinente sur la *détection de changement* dans la BBA à filtrer. En effet, si la variation des croyances sur les éléments focaux de la BBA à filtrer est très différente de la variation des

[4]Dans [Sme93], il est montré que l'affaiblissement peut être formalisé en utilisant le GBT.

croyances sur les éléments focaux de la BBA prédite, cela peut signifier que le modèle de prédiction n'est plus adapté. Un changement dans la BBA mesurée induit donc du conflit car la prédiction et la mesure sont discordantes, et la valeur et la durée du conflit dépendent de la rapidité du changement[5]. Bien qu'il soit important d'exploiter le conflit, il est tout aussi important de le gérer (cf. chapitre 2) pour éviter de propager son effet absorbant lors du traitement de fonctions de croyance temporelles.

### 3.2.1 Définition des concepts utilisés

**Croyances binaires** Nous considérons des fonctions de croyance temporelles *binaires*, c'est à dire définies sur un cadre de discernement à deux hypothèses $\Omega_t = \{T_t, F_t\}$ où $t$ est l'instant, $T_t$ signifie *vrai* (par exemple l'état est *vrai* ou la classe est *vraie*) et $F_t$ signifie *faux*. Un exemple d'évolution de BBA binaire sur un intervalle temporel $[1, 1168]$ est donné sur la figure 3.1 où l'hypothèse $T_t$ signifie qu'« il existe un mouvement de caméra » et l'hypothèse $F_t$ signifie qu'« il n'existe pas de mouvement de caméra » dans la vidéo étudiée.

**Notion d'état** Les croyances à filtrer reflètent la *véracité* d'un état d'un système observé. On dit que *l'état est vrai* si la pignistique de l'état $T_t$ est la plus élevée, c'est à dire $\mathbf{BetP}(T_t) > \mathbf{BetP}(F_t)$, et *dans l'état faux* sinon.

**Notion de modèle d'évolution** Nous proposons d'utiliser deux modèles lors du calcul de la prédiction : un *modèle « état vrai »* que nous notons $\mathcal{T}$ et un *modèle « état faux »* noté $\mathcal{F}$. Ces modèles $\mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$ sont utilisés alternativement en fonction de la véracité de l'état.

**Notion de commutation de modèle** Si le modèle $\mathcal{T}$ (resp. $\mathcal{F}$) était utilisé à $t$ et si un changement est détecté dans la BBA à filtrer à cet instant alors le modèle est changé et le modèle $\mathcal{F}$ (resp. $\mathcal{T}$) sera utilisé à $t+1$. On parle alors de commutation de modèle. Une commutation de modèle intervient lorsque le conflit entre la prédiction et la mesure est trop important.

### 3.2.2 Les concepts sur un exemple : segmentation de vidéos

Nous cherchons dans cet exemple à illustrer l'utilité du filtre temporel crédibiliste sur un exemple de segmentation de vidéo en fonction du mouvement de caméra. Cet exemple introduit par ailleurs les notations de base utilisées dans ce chapitre ainsi que quelques concepts.

La figure 3.1a illustre l'évolution de la BBA en sortie d'une source de croyance générée à partir d'observations. Cette BBA indique si la caméra bouge ($T_t$) ou ne bouge pas ($F_t$) à l'instant $t$ et le mouvement est évalué à l'image $t$ indépendamment de celui évalué sur les autres images. Le cadre de discernement est donc binaire et égal à :

$$\Omega_t = \{T_t, F_t\} \tag{3.1}$$

Par extension, on dira que la fonction de croyance est elle-aussi binaire.

A chaque instant $t \in \{1 \dots 1168\}$ correspondants aux $1168$ images de la vidéo, une fonction de croyance $m^{\Omega_t}$ est fournie par la source :

[5]Cette propriété est intéressante pour la surveillance de systèmes dynamiques comme des procédés industriels par exemple.

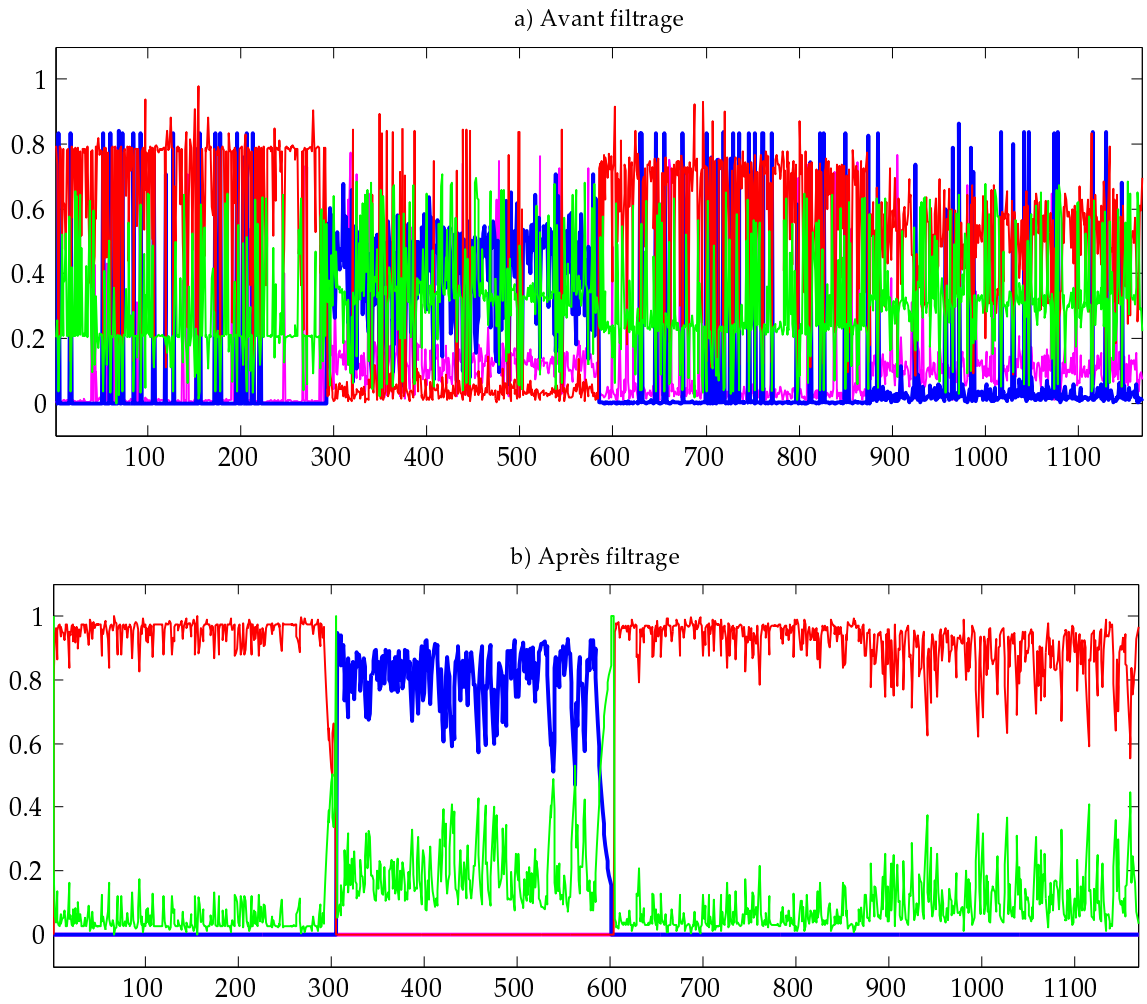



**FIG. 3.1 :** *Un résultat de filtrage de fonctions de croyance temporelles bruitées par le TCF proposé.*

- La croyance sur le fait que la caméra bouge à $t$ est $m^{\Omega_t}(T_t)$ et porte la couleur *bleue*,
- La croyance sur le fait que la caméra ne bouge pas à $t$ est $m^{\Omega_t}(F_t)$ et porte la couleur *rouge*,
- La croyance sur l'ignorance concernant le mouvement à $t$ est $m^{\Omega_t}(\{F_t, T_t\}) = m^{\Omega_t}(\Omega_t)$ et porte la couleur *verte*,
- La croyance sur le fait que la caméra fasse autre chose que bouger ou ne pas bouger à $t$ est $m^{\Omega_t}(\emptyset_t)$ et porte la couleur *magenta*.

Notons que, tout au long de ce chapitre, nous utiliserons fréquemment des fonctions de croyance binaires et les couleurs des traits sur les figures respecteront les couleurs utilisées ci-dessus.

La figure 3.1a montre une croyance temporelle *chahutée* : la croyance *varie de façon importante* entre deux instants consécutifs et de nombreuses *fausses alarmes* apparaissent. On sait identifier les fausses alarmes par la brièveté des mouvements de caméra détectés, ces derniers ne pouvant durer que quelques images. On s'aperçoit que l'intervalle $[300, 600]$ correspond à un mouvement de caméra.

Le filtre temporel TCF est appliqué sur l'ensemble de ces croyances temporelles en émettant l'hypothèse que la caméra ne peut bouger de façon importante entre deux images successives. La

figure 3.1b montre le résultat du filtrage par le TCF proposé dans ce chapitre. En plus d'être filtrée, la distribution de croyances obtenue en sortie du TCF a les caractéristiques suivantes :

- **Consonante :** A chaque instant $t$, le filtre génère une fonction de croyance composé d'au plus deux éléments focaux :
  - Il existe un mouvement de caméra $\Rightarrow$ les éléments focaux sont $m^{\Omega_t}(T_t)$ et $m^{\Omega_t}(\Omega_t)$,
  - Il n'existe pas de mouvement de caméra $\Rightarrow$ les éléments focaux sont $m^{\Omega_t}(F_t)$ et $m^{\Omega_t}(\Omega_t)$.
- **Normale :** A chaque instant $t$, le filtre génère une fonction de croyance sans conflit, c'est à dire $m^{\Omega_t}(\emptyset_t) = 0$.

## 3.3 Description du filtre temporel crédibiliste

L'objectif du filtre est de déterminer la sortie $m^{\Omega_t}$ à l'instant courant $t$ à partir de la mesure $\tilde{m}^{\Omega_t}$ à l'instant courant, de la sortie du filtre $m^{\Omega_{t-1}}$ à l'instant précédent $t-1$ et d'un modèle d'évolution $\mathcal{M}$ de la BBA de sortie du filtre entre deux les instants successifs $t$ et $t-1$. On note :

| | |
|---|---|
| - $\hat{m}^{\Omega_t}$ | la BBA prédite à l'instant $t$, |
| - $\mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$ | le modèle d'évolution courant de la BBA de sortie du filtre entre deux instants et le modèle peut être soit $\mathcal{T}$ soit $\mathcal{F}$, |
| - $m^{\Omega_{t-1}}$ | la sortie du filtre à l'instant précédent $t-1$, |
| - $\tilde{m}^{\Omega_t}$ | la BBA des observations (ou encore les *mesures*) à $t$. |

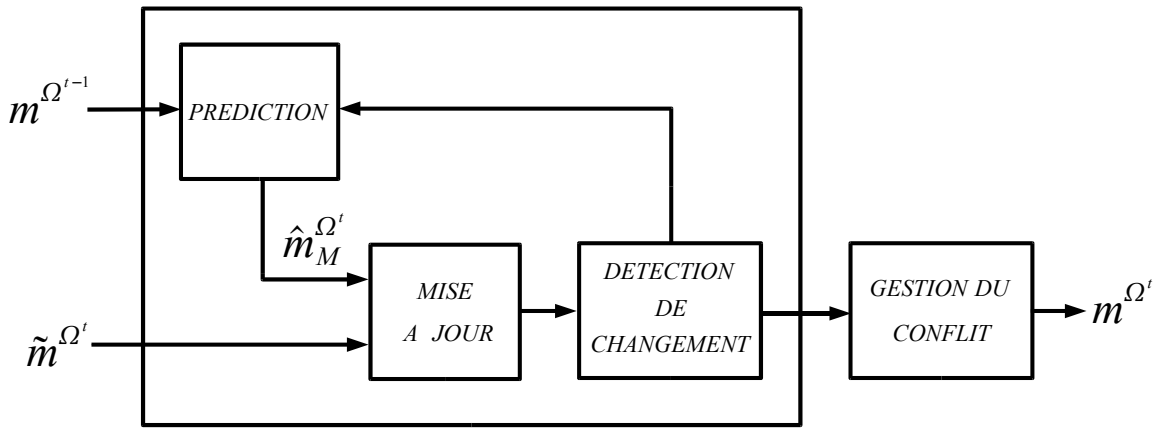



**FIG. 3.2 :** *Principe du filtre temporel crédibiliste TCF.*

### 3.3.1 Principe du filtre

Le principe du filtre temporel crédibiliste est décrit Fig. 3.2. La prédiction de la BBA courante $\hat{m}^{\Omega_t}$ est déterminée à partir de la sortie du filtre à l'instant précédent $m^{\Omega_{t-1}}$ et du modèle d'évolution courant $\mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$. La prédiction obtenue est ensuite combinée avec la BBA mesurée $\tilde{m}^{\Omega_t}$ provenant des observations à l'instant $t$. La combinaison de la prédiction avec la mesure est réalisée avec un opérateur conjonctif permettant d'obtenir une BBA plus spéficique que les deux BBA combinées.

L'opérateur conjonctif, dans le cadre du TBM, met en valeur le conflit correspondant dans notre cas à l'inadéquation entre le modèle utilisé et les données en cours.

Nous étudierons successivement les différents blocs de la figure 3.2 à savoir la prédiction, la mise à jour, la détection de changement et la gestion du conflit.

### 3.3.2 Prédiction

Sous l'hypothèse de Markovianité[6], le modèle d'évolution lie la BBA de sortie du filtre entre deux instants consécutifs, $t$ et $t-1$. Dans le cadre du TBM, nous traduisons cette dépendance entre les BBAs à deux instants consécutifs par des BBA conditionnelles $m^{\Omega_t}[\omega_{t-1}]$ définies sur l'espace $\Omega_t$ de l'instant courant conditionnellement aux sous-ensembles $\omega_{t-1} \in 2^{\Omega_{t-1}} = \{\emptyset_{t-1}, T_{t-1}, F_{t-1}, \{T_{t-1}, F_{t-1}\}\}$ de l'instant précédent.

Le filtre proposé utilise deux modèles d'évolution : un premier modèle ($\mathcal{T}$) est appliqué lorsque les croyances indiquent que l'état est globalement *vrai* et un second modèle ($\mathcal{F}$) est appliqué lorsque les croyances indiquent que l'état est globalement *faux* :

- **Modèle $\mathcal{T}$ :** l'hypothèse courante de l'état est $T_t$, c'est à dire l'état est vrai.
  Dans ce cas le modèle est une règle implicative :
  « *si l'état à $t-1$ était $T_{t-1}$ alors l'état à $t$ sera $T_t$ avec une croyance $\gamma_{\mathcal{T}} \in [0,1]$* »
- **Modèle $\mathcal{F}$ :** l'hypothèse courante de l'état est $F_t$, c'est à dire l'état est faux.
  Dans ce cas le modèle est une règle implicative :
  « *si l'état à $t-1$ était $F_{t-1}$ alors l'état à $t$ sera $F_t$ avec une croyance $\gamma_{\mathcal{F}} \in [0,1]$* »

Le modèle $\mathcal{T}$ (resp. $\mathcal{F}$) explicite le fait que si l'état était vrai (resp. faux) à un instant alors il sera encore vrai (resp. faux) à l'instant suivant. Chaque modèle est caractérisé par un paramètre $\gamma_{\mathcal{M}} \in [0,1], \mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$ et nous proposerons une méthode d'estimation dans le paragraphe 3.5.

Les modèles apparaissent comme des règles implicatives (ces règles ont été décrites dans § 2.2.3.1). L'écriture de ces règles à l'aide de BBA conditionnelles est intuitive et prend la forme suivante :

$$\text{Formes conditionnelles} = \begin{cases} \textbf{Modèle « l'état est vrai »} \\ \mathcal{T} : m^{\Omega_t}_{\mathcal{T}}[T_{t-1}](T_t) = \gamma_{\mathcal{T}} \\ \quad m^{\Omega_t}_{\mathcal{T}}[T_{t-1}](\Omega_t) = 1 - \gamma_{\mathcal{T}} \\ \\ \textbf{Modèle « l'état est faux »} \\ \mathcal{F} : m^{\Omega_t}_{\mathcal{F}}[F_{t-1}](F_t) = \gamma_{\mathcal{F}} \\ \quad m^{\Omega_t}_{\mathcal{F}}[F_{t-1}](\Omega_t) = 1 - \gamma_{\mathcal{F}} \end{cases} \tag{3.2}$$

Le GBT (Eq. 2.28) permet de calculer la BBA prédite $\hat{m}^{\Omega_t}_{\mathcal{M}}$ à $t$ en fonction de la BBA de sortie du filtre à $t-1$ et les masses du modèle en cours $\mathcal{M}$ définies pour $t$ conditionnellement à $t-1$ (Eq. 2.28). Cependant, le calcul par le GBT nécessite de déterminer le conditionnement par rapport à chaque proposition $B \in 2^{\Omega_t} = \{\emptyset_t, \{T_t\}, \{F_t\}, \{T_t, F_t\}\}$ quel que soit le modèle mais l'équation 3.2 ne définit que partiellement le modèle. Il manque à définir deux BBA conditionnelles :

[6]Cette hypothèse est très répandue en traitement de données temporelles [Rab89, Mur02, Sme05a, RS04] afin de réduite les coûts de calcul et l'espace mémoire.

– $m_{\mathcal{T}}^{\Omega_t}[F_{t-1}]$ : traduit la connaissance sur les propositions $B \in 2^{\Omega_t}$ à l'instant courant sous l'hypothèse que *l'état était faux* à $t-1$ et que le modèle utilisé à $t$ est « $\mathcal{T}$ *: l'état est vrai* ».
$\Rightarrow$ Nous choisissons une attitude conservatrice en optant pour une BBA vide traduisant notre méconnaissance sur ce cas :

$$m_{\mathcal{T}}^{\Omega_t}[F_{t-1}](\Omega_t) = 1 \tag{3.3}$$

– $m_{\mathcal{F}}^{\Omega_t}[T_{t-1}]$ : traduit la connaissance sur les propositions $B \in 2^{\Omega_t}$ à l'instant courant sous l'hypothèse que *l'état était vrai* à $t-1$ et que le modèle utilisé à $t$ est « $\mathcal{F}$ *: l'état est faux* ».
$\Rightarrow$ Nous choisissons une attitude conservatrice en optant pour une BBA vide traduisant notre méconnaissance sur ce cas :

$$m_{\mathcal{F}}^{\Omega_t}[T_{t-1}](\Omega_t) = 1 \tag{3.4}$$

Sous l'hypothèse que les BBA conditionnelles sont toutes distinctes[7], les BBA $m_{\mathcal{T}}^{\Omega_t}[\Omega_{t-1}]$ et $m_{\mathcal{F}}^{\Omega_t}[\Omega_{t-1}]$ sont déduites par application de la DRC (Eq. 2.29) :

$$\begin{array}{rcl} m_{\mathcal{T}}^{\Omega_t}[\Omega_{t-1}] & = & m_{\mathcal{T}}^{\Omega_t}[T_{t-1}] \textcircled{\cup} m_{\mathcal{T}}^{\Omega_t}[F_{t-1}] \\ m_{\mathcal{F}}^{\Omega_t}[\Omega_{t-1}] & = & m_{\mathcal{F}}^{\Omega_t}[T_{t-1}] \textcircled{\cup} m_{\mathcal{F}}^{\Omega_t}[F_{t-1}] \end{array} \tag{3.5}$$

Cependant, l'attitude conservatrice dans le choix de l'affectation des masses dans les équations 3.3 et 3.4 implique que la DRC va générer une BBA vide. En effet, $m_{\mathcal{T}}^{\Omega_t}[F_{t-1}](\Omega_t) = 1$ et $m_{\mathcal{F}}^{\Omega_t}[T_{t-1}](\Omega_t) = 1$ sont absorbants par la DRC. On obtient donc :

$$\begin{array}{rcl} m_{\mathcal{F}}^{\Omega_t}[\Omega_{t-1}] & = & 1 \\ m_{\mathcal{F}}^{\Omega_t}[\Omega_{t-1}] & = & 1 \end{array} \tag{3.6}$$

Le modèle de prédiction est composé de 4 BBA conditionnelles, c'est à dire autant que de sous-ensembles d'hypothèses dans $2^{\Omega_{t-1}}$, et dont 2 sont vides :

$$\begin{array}{lcl} & \textbf{Modèle « l'état est vrai »} & \\ \mathcal{T} & : & m_{\mathcal{T}}^{\Omega_t}[T_{t-1}](T_t) = \gamma_{\mathcal{T}} \\ & & m_{\mathcal{T}}^{\Omega_t}[T_{t-1}](\Omega_t) = 1-\gamma_{\mathcal{T}} \\ & & m_{\mathcal{T}}^{\Omega_t}[F_{t-1}](\Omega_t) = 1 \\ & & m_{\mathcal{T}}^{\Omega_t}[\Omega_{t-1}](\Omega_t) = 1 \\ & & m_{\mathcal{T}}^{\Omega_t}[\emptyset_{t-1}](\emptyset_t) = 1 \end{array} \qquad \begin{array}{lcl} & \textbf{Modèle « l'état est faux »} & \\ \mathcal{F} & : & m_{\mathcal{F}}^{\Omega_t}[F_{t-1}](F_t) = \gamma_{\mathcal{F}} \\ & & m_{\mathcal{F}}^{\Omega_t}[F_{t-1}](\Omega_t) = 1-\gamma_{\mathcal{F}} \\ & & m_{\mathcal{F}}^{\Omega_t}[T_{t-1}](\Omega_t) = 1 \\ & & m_{\mathcal{F}}^{\Omega_t}[\Omega_{t-1}](\Omega_t) = 1 \\ & & m_{\mathcal{F}}^{\Omega_t}[\emptyset_{t-1}](\emptyset_t) = 1 \end{array} \tag{3.7}$$

Le GBT (Eq. 2.28) permet d'obtenir la prédiction :

$$\hat{m}_{\mathcal{M}}^{\Omega_t}(C) = \sum_{B \subseteq \Omega_{t-1}} m_{\mathcal{M}}^{\Omega_t}[B](C) \cdot m^{\Omega_{t-1}}(B), \forall C \subseteq \Omega_t \tag{3.8}$$

On pose comme hypothèse que la sortie du filtre est *consonante* et *normale* (l'ensemble vide n'est pas un élément focal) avec au plus deux éléments focaux :
– $T_t$ et $\Omega_t$ si le modèle en cours est $\mathcal{T}$,
– $F_t$ et $\Omega_t$ si le modèle en cours est $\mathcal{F}$.

[7] Sinon il suffit d'appliquer la règle BDRC (Eq. 2.65).

Par conséquent, la BBA conditionnelle $m^{\Omega_t}_{\mathcal{M}}[\emptyset_{t-1}]$ n'a pas d'impact sur la BBA prédite. De plus, du fait du choix des modèles de prédiction, la BBA prédite est donc elle-aussi consonante avec les mêmes éléments focaux que la sortie du filtre à $t-1$. En fonction des modèles d'évolution $\mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$, les éléments focaux de la BBA prédite sont donnés par l'équation 3.8 :

$$\begin{array}{rcl} \hat{m}^{\Omega_t}_{\mathcal{T}}(T_t) & = & \gamma_{\mathcal{T}} \cdot m^{\Omega_{t-1}}(T_{t-1}) \\ \hat{m}^{\Omega_t}_{\mathcal{T}}(\Omega_t) & = & \gamma_{\mathcal{T}} \cdot m^{\Omega_{t-1}}(\Omega_{t-1}) + 1 - \gamma_{\mathcal{T}} \end{array} \qquad \begin{array}{rcl} \hat{m}^{\Omega_t}_{\mathcal{F}}(F_t) & = & \gamma_{\mathcal{F}} \cdot m^{\Omega_{t-1}}(F_{t-1}) \\ \hat{m}^{\Omega_t}_{\mathcal{F}}(\Omega_t) & = & \gamma_{\mathcal{F}} \cdot m^{\Omega_{t-1}}(\Omega_{t-1}) + 1 - \gamma_{\mathcal{F}} \end{array} \tag{3.9}$$

La prédiction est donc consonante et les mêmes éléments focaux sont identiques à ceux de la sortie du filtre à $t-1$. De plus, on remarque que l'application du modèle d'évolution (Eq. 3.8) revient en fait à un *affaiblissement* (Eq. 2.13) où $\gamma_{\mathcal{M}}$ peut être interprété comme la *fiabilité d'un couple d'hypothèses à deux instants consécutifs*. Cette propriété hérite de la particularité du modèle employé et de la redistribution du conflit.

### 3.3.3 Mise à jour

Prédiction $\hat{m}^{\Omega_t}_{\mathcal{M}}$ et mesure $\tilde{m}^{\Omega_t}$ représentent deux sources d'information concernant l'état. La combinaison conjonctive de ces distributions de masse permet de renforcer le choix du modèle en cours (spécialisation de la prédiction) :

$$m^{\Omega_t} = \hat{m}^{\Omega_t}_{\mathcal{M}} \textcircled{\cap} \tilde{m}^{\Omega_t} \tag{3.10}$$

La CRC permet de mettre en évidence le degré de conflit entre la prédiction à $t$ et les données courantes. Le conflit joue le rôle d'un indicateur de changement de modèle. En effet, le conflit apparaît lorsque le modèle utilisé dans le calcul de la prédiction n'est plus adapté aux données courantes. Sa valeur $\epsilon_t$ est donnée par la masse sur l'ensemble vide après combinaison conjonctive :

$$\epsilon_t = (\hat{m}^{\Omega_t}_{\mathcal{M}} \textcircled{\cap} \tilde{m}^{\Omega_t})(\emptyset_t) \tag{3.11a}$$

$$= \hat{m}^{\Omega_t}_{\mathcal{M}}(\emptyset_t) + \tilde{m}^{\Omega_t}(\emptyset_t) + \hat{m}^{\Omega_t}_{\mathcal{M}}(T_t) \cdot \tilde{m}^{\Omega_t}(F_t) + \hat{m}^{\Omega_t}_{\mathcal{M}}(F_t) \cdot \tilde{m}^{\Omega_t}(T_t) \tag{3.11b}$$

On a vu dans le paragraphe précédent que ni le modèle d'évolution ni la prédiction ne présentait de conflit interne. Par conséquent :

$$\hat{m}^{\Omega_t}(\emptyset_t) = \hat{m}^{\Omega_t}_{\mathcal{F}}(T_t) = \hat{m}^{\Omega_t}_{\mathcal{T}}(F_t) = 0$$

L'expression du conflit $\epsilon_t$ se réduit donc à la somme de deux éléments : le conflit provenant de la fusion des caractéristiques mesurées $\tilde{m}^{\Omega_t}(\emptyset_t)$ et le conflit lié à la contradiction entre modèle et mesure. On obtient donc :

$$\epsilon_t = \begin{cases} \tilde{m}^{\Omega_t}(\emptyset_t) + \hat{m}^{\Omega_t}_{\mathcal{T}}(T_t) \cdot \tilde{m}^{\Omega_t}(F_t) & \text{si le modèle en cours est } \mathcal{T} \\ \tilde{m}^{\Omega_t}(\emptyset_t) + \hat{m}^{\Omega_t}_{\mathcal{F}}(F_t) \cdot \tilde{m}^{\Omega_t}(T_t) & \text{si le modèle en cours est } \mathcal{F} \end{cases} \tag{3.12}$$

Pour détecter un changement de modèle, le conflit $\epsilon_t$ ne peut pas être utilisé directement car il peut provenir des erreurs de mesures sur les caractéristiques (cf. Fig. 3.3c) sans correspondre réellement à un changement d'état. Afin d'éviter les changements d'état intempestifs, nous avons utilisé un processus de somme cumulée (CUSUM) pondérée décrit dans la section suivante.

### 3.3.4 Détection de changement d'état

La CUSUM est la somme cumulée pondérée de la masse de conflit. La détection de changement de modèle est basé sur un simple seuillage de la valeur de la CUSUM. Lorsque le seuil $\mathbb{T}_s$ (**s**top) est atteint à l'instant $t_s$, le modèle d'évolution est changé. Le nouveau modèle est appliqué à partir de $t_s$.

Pour gérer les conflits de faible amplitude et de longue durée ou encore les conflits de forte amplitude et de faible durée, nous proposons d'utiliser un mécanisme d'*oubli* sur la CUSUM courante $\mathbf{CS}(t)$ :

$$\mathbf{CS}(t) \leftarrow \mathbf{CS}(t-1) \cdot \lambda + \epsilon_t \tag{3.13}$$

Le coefficient d'oubli $\lambda \in [0, 1]$ a été choisi constant et il est appliqué à chaque instant. La détection de changement est basé sur un seuillage de la valeur de la somme cumulée. Lorsque le seuil $\mathbb{T}_s$ (**s**top) est atteint à l'instant $t_s$, le modèle d'évolution est changé. Le nouveau modèle est appliqué à partir de l'instant $t_s$.

**Exemple 16.** *La figure 3.3 illustre un résultat du filtrage correspondant à la détection d'une action* sauter *dans un saut en hauteur (la vérité terrain apparaît surlignée en trait noir et gras sur l'axe temporel). La croyance à filtrer (Fig. 3.3a), c'est à dire la mesure, est bruitée, présente des fausses alarmes ainsi que du conflit.*

*Le résultat du filtrage (Fig. 3.3b), avec un réglage adapté aux données (seuil de commutation de modèle fixé à* $2.2$*, cf. ligne pointillée en 3.3d), montre le rôle du processus de rupture de modèle : la première rupture apparaît vers* $t = 47$ *et la seconde vers* $t = 63$*. A partir de* $t = 1$*, le modèle « l'état est faux » est appliqué. Le critère de détection de changement (Fig. 3.3d) indique qu'à* $t = 49$ *le modèle doit être remplacé car il ne correspond plus aux données qui ont évolué. Le modèle « l'état est vrai » est appliqué à partir de la première rupture (une zone de doute est imposée de* $t = 47$ *à* $t = 49$*). Puis à* $t = 63$*, une seconde rupture a lieu : l'état redevient faux.*

*La figure 3.3c correspond au conflit entre le modèle et les données. Les chahutements proviennent des variations dans la mesure impliquant des discordances entre prédiction et mesure.*

*La figure 3.3d illustre le critère de décision correspondant à la somme cumulée pondérée du conflit. La pondération permet d'oublier progressivement les fausses détections (cf. la décroissance sur* $[30, 45]$*). Notons qu'un décalage existe entre les croyances et la somme cumulée car à l'issue d'un changement de modèle, un retour en arrière de quelques instants est effectué.*

A l'instant $t$, seul l'un des deux modèles ($\mathcal{T}$ et $\mathcal{F}$) est appliqué. Lorsqu'un changement de modèle doit intervenir pour l'état concerné, c'est à dire le seuil d'arrêt $\mathbb{T}_s$ est atteint par la CUSUM, l'intervalle de temps noté $\mathbf{I}_d$ tel que :

$$\mathbf{I}_d = [t_s - \mathcal{W}, t_s] \tag{3.14}$$

est interprété comme un intervalle de transition entre les deux modèles d'évolution de l'état. Le paramètre $\mathcal{W}$ permet de limiter la taille de la transition. Une distribution de masse modélisant l'ignorance est affectée en sortie du filtre sur tout l'intervalle $\mathbf{I}_d$ :

$$m_{\mathbf{I}_d}^{\Omega_t}(\Omega_t) = 1 \tag{3.15}$$

Puis, à partir de la borne supérieure de $\mathbf{I}_d$ (c'est à dire $t_s$), le nouveau modèle (en concordance avec les mesures) est appliqué et la CUSUM est remise à zéro.

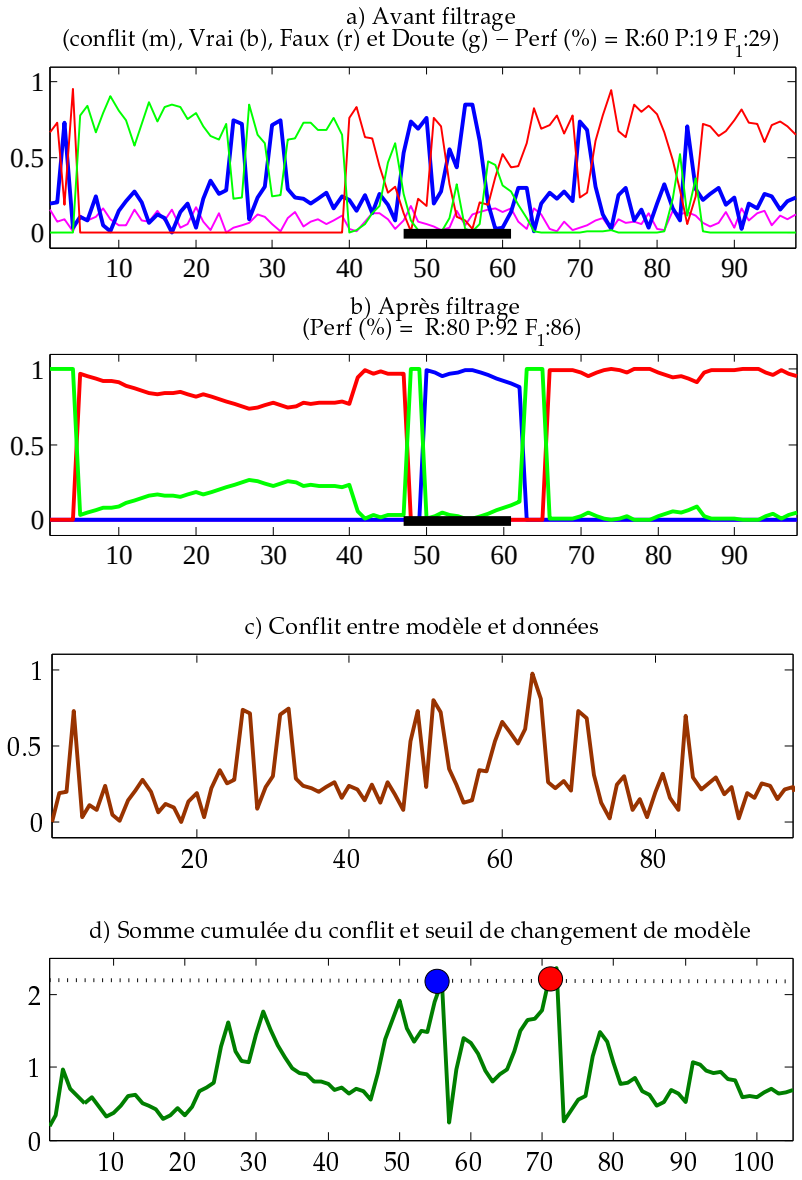



**FIG. 3.3 :** *Un résultat de filtrage.*

### 3.3.5 Sortie du filtre : gestion du conflit

Peu avant le dépassement du seuil (phase transitoire du changement de modèle), la BBA issue de la combinaison entre prédiction et mesure peut faire apparaître un conflit élevé. Or, nous souhaitons que la BBA de sortie du filtre soit consonante et sans conflit. En conséquence, la masse de conflit $\epsilon_t$ doit être redistribuée. Dans cette section, nous présentons les mécanismes de formatage de la sortie du filtre afin d'obtenir une BBA consonante.

#### 3.3.5.1 Importance du conflit

La mise à jour de la BBA courante de sortie du filtre dépend de deux cas de figure :

– Lorsque la valeur instantanée du conflit ($\epsilon_t$) entre la prédiction et la mesure est importante, nous

faisons confiance au modèle d'évolution en conservant la BBA prédite :

$$m^{\Omega_t} = \hat{m}^{\Omega_t}_{\mathcal{M}} \quad \text{si} \quad \epsilon_t \geq \delta_\emptyset \tag{3.16}$$

Le seuil de tolérance au conflit instantané est adapté automatiquement par une moyenne de la valeur du conflit sur un intervalle de temps ($N$) :

$$\delta_\emptyset = \frac{1}{N} \sum_{t_i=(t-N-1)}^{t} \epsilon_{t_i} \tag{3.17}$$

où $N$ est inférieur à la durée de l'état en cours. Ce seuillage est bien adapté lorsque la distribution de masses à filtrer présente du conflit. Lorsque le conflit est constamment présent mais sans pour autant indiquer un changement dans la distribution de masses à filtrer, alors ce seuillage permet de s'affranchir des commutations.

– Lorsque la valeur instantanée du conflit ($\epsilon_t$) entre la prédiction et la mesure est faible ($< \delta_\emptyset$), la sortie du filtre est mise à jour en calculant la combinaison conjonctive de la prédiction avec la mesure :

$$m^{\Omega_t} = \hat{m}^{\Omega_t}_{\mathcal{M}} \textcircled{\cap} \tilde{m}^{\Omega_t} \quad \text{si} \quad \epsilon_t < \delta_\emptyset \tag{3.18}$$

où le seuil de tolérance est donné par l'équation 3.17.

On peut remarquer que dans le cas où la masse de conflit instantanée $\epsilon_t$ est supérieure au seuil de tolérance $\delta_\emptyset$ à des instants successifs $t_0, t_1, t_2 \ldots t$, la distribution de masses en sortie du filtre $m^{\Omega_t}$ à ces instants correspondra toujours à la prédiction. Par conséquent, vue la forme du modèle de prédiction, la sortie du filtre tendra vers la BBA *vide*, c'est à dire $\lim_{t\to\infty} m^{\Omega_t}(\Omega_t) \to 1$, ce qui reflète l'ignorance totale (logiquement car ceci correspond au cas où le modèle n'est plus en accord avec les données depuis trop longtemps). La taux de convergence de la distribution de masse initiale $m^{\Omega_{t=t_0}}$ vers la distribution vide dépend de la valeur de $\gamma_{\mathcal{M}}$. Il est possible d'estimer la durée nécessaire pour qu'une BBA simple telle que $m^{\Omega_{t_0}}(T_{t_0}) = v_0$ tende vers $m^{\Omega_t}(T_t) = v_1$ ($v_1 < v_0$) et cette durée vaut $\tau = \frac{\log(v_1/v_0)}{\log(\gamma_{\mathcal{M}})}$. Ainsi, après $\tau$ dépassements consécutifs de la valeur du seuil de tolérance du conflit $\delta_\emptyset$, la valeur de la masse sur l'ignorance sera de $m^{\Omega_t}(\Omega_t) = 1 - v_1$.

#### 3.3.5.2 Gestion du conflit interne

Le conflit $\epsilon_t$ entre prédiction et mesure doit être redistribué. De plus, nous souhaitons que si le modèle actif est $\mathcal{T}$ alors les éléments focaux sont $T_t$ et $\Omega_t$, et si le modèle actif est $\mathcal{F}$ alors les éléments focaux sont $F_t$ et $\Omega_t$.

Nous présentons la gestion du conflit lorsque le modèle actif est $\mathcal{T}$, la généralisation au cas où le modèle actif est $\mathcal{F}$ étant très rapide. Dans le cas de $\mathcal{T}$ on applique la redistribution suivante :

– **Redistribution de la masse de conflit** $m^{\Omega_t}(\emptyset_t)$ **–** Cette masse de conflit est affectée à $T_t$, c'est à dire sur l'hypothèse soutenue par le modèle actif. Ce conflit est égal à la somme du conflit provenant de la fusion des caractéristiques ($\tilde{m}^{\Omega_t}(\emptyset_t)$) et de la contradiction entre l'hypothèse $T_t$ du modèle et l'hypothèse $F_t$ des observations (contradiction quantifiée par $\hat{m}^{\Omega_t}_{\mathcal{T}}(T_t) \cdot \tilde{m}^{\Omega_t}(F_t)$). La masse $\tilde{m}^{\Omega_t}(\emptyset_t)$ représente un conflit non résolu avant filtrage et nous supposons qu'il est accepté par l'utilisateur [Sme05a] tandis que la masse $\hat{m}^{\Omega_t}_{\mathcal{T}}(T_t) \cdot \tilde{m}^{\Omega_t}(F_t)$ prend en compte la

véracité du modèle d'évolution en cours auquel nous faisons confiance.

– **Redistribution de la masse** $m^{\Omega_t}(F_t)$ **–** Cette masse de conflit est affectée à $\Omega_t$, c'est à dire sur le doute concernant le modèle utilisé. L'élément $F_t$ est en conflit avec le modèle $\mathcal{T}$ et reflète un changement potentiel de modèle. Il n'est donc pas transféré sur l'hypothèse en cours ($T_t$) car il pourrait induire du conflit encore plus important à l'instant suivant (effet « catalyseur »).

Le même raisonnement tient lorsque le modèle est $\mathcal{F}$. Les redistributions sont résumées comme suit :

**Redistribution lorsque le modèle est « $\mathcal{T}$ : l'état est vrai »**

$$\begin{aligned} m^{\Omega_t}(T_t) &\leftarrow m^{\Omega_t}(T_t) + m^{\Omega_t}(\emptyset_t) \\ m^{\Omega_t}(\Omega_t) &\leftarrow m^{\Omega_t}(\Omega_t) + m^{\Omega_t}(F_t) \\ m^{\Omega_t}(\emptyset_t) &= m^{\Omega_t}(F_t) = 0 \end{aligned} \tag{3.19}$$

**Redistribution lorsque le modèle est « $\mathcal{F}$ : l'état est faux »**

$$\begin{aligned} m^{\Omega_t}(F_t) &\leftarrow m^{\Omega_t}(F_t) + m^{\Omega_t}(\emptyset_t) \\ m^{\Omega_t}(\Omega_t) &\leftarrow m^{\Omega_t}(\Omega_t) + m^{\Omega_t}(T_t) \\ m^{\Omega_t}(\emptyset_t) &= m^{\Omega_t}(T_t) = 0 \end{aligned} \tag{3.20}$$

## 3.4 Critère de qualité locale

Nous proposons de définir un critère permettant de mesurer la qualité de l'état filtré. On note ce critère $LQ$ (*local quality*). A un instant donné $t$, la qualité de l'état filtré est élevée si le degré de confiance sur la réalisation de l'état est élevée et si le conflit est faible. Le degré de confiance peut être quantifié par la probabilité pignistique, la croyance ou encore la plausibilité sur $T_t$ ou $F_t$.

Sur l'intervalle de temps où un modèle $\mathcal{M}$ est utilisé, nous calculons la moyenne des degrés de confiance pondérée par l'opposé du conflit. Par exemple, si la confiance est représentée par une probabilité pignistique alors le critère est défini récursivement de la façon suivante :

$$LQ^{\mathrm{BetP}}_{t_s:t}(h) = \left(1 - \frac{1}{t - t_s}\right) \cdot LQ^{\mathrm{BetP}}_{t_s:(t-1)}(h) + \frac{\mathbf{BetP}^{\Omega_t}(h)}{t - t_s} \cdot (1 - \epsilon_t), \forall h \in \Omega_t \tag{3.21}$$

Cette définition du critère correspond à une moyenne calculée en ligne qui intègre la qualité passée et le degré de confiance sur l'état en cours pondéré par l'opposé du conflit. La variable $h$ est égale à $T_t$ si le modèle utilisé est $\mathcal{T}$ et vaut $F_t$ si le modèle est $\mathcal{F}$. De la même manière on peut définir $LQ^{bel}_{t_s:t}$ et $LQ^{pl}_{t_s:t}$ vérifiant $LQ^{\mathrm{BetP}}_{t_s:t}, LQ^{bel}_{t_s:t}, LQ^{pl}_{t_s:t} \in [0, 1]$ et :

$$LQ^{bel}_{t_s:t} \leq LQ^{\mathrm{BetP}}_{t_s:t} \leq LQ^{pl}_{t_s:t} \tag{3.22}$$

Ce critère est borné sur $[0, 1]$ car les degrés de confiance $bel$, $pl$ et $\mathbf{BetP}$ sont bornés sur $[0, 1]$ et donc la moyenne l'est aussi. Dans le cas d'un espace de discernement binaire et pour l'exemple de la probabilité pignistique, le critère $LQ^{\mathrm{BetP}}_{t_s:t}$ peut s'écrire :

$$\begin{aligned} LQ^{\mathrm{BetP}}_{t_s:t}(T_t) &= \left(1 - \frac{1}{t - t_s}\right) \cdot LQ^{\mathrm{BetP}}_{t_s:(t-1)}(T_t) + \frac{m^{\Omega_t}(T_t) + 0.5 \cdot m^{\Omega_t}(T_t \cup F_t)}{t - t_s} \\ LQ^{\mathrm{BetP}}_{t_s:t}(F_t) &= \left(1 - \frac{1}{t - t_s}\right) \cdot LQ^{\mathrm{BetP}}_{t_s:(t-1)}(F_t) + \frac{m^{\Omega_t}(F_t) + 0.5 \cdot m^{\Omega_t}(T_t \cup F_t)}{t - t_s} \end{aligned} \tag{3.23}$$

où le facteur $(1 - \epsilon_t)$ est annulé du fait de la définition de la transformée pignistique.

**Exemple 17.** *Un exemple de qualité évoluant au cours du temps (correspondant aux données de la figure 3.3) avec la probabilité pignistique en guise de degré de confiance est donné sur la figure 3.4. Les grandes variations sont dues aux réinitialisations de la qualité lors des changements de modèles. Dans cet exemple, on remarque que la qualité du filtrage avec le modèle $\mathcal{T}$ (sur l'intervalle $[49, 63]$, figure de gauche) est environ égale à $0.51$ (valeur en $t = 63$). La qualité du filtrage avec le modèle* état vrai *n'est pas élevée à cause du conflit sur les données initiales (Fig. 3.3a). Le filtrage avec le modèle $\mathcal{F}$ (état faux) mène à $LQ^{1:47}_{\text{BetP}}(F_t) = 0.65$ et $LQ^{66:98}_{\text{BetP}}(F_t) = 0.77$, soit une qualité moyenne de $0.71$. On peut en conclure que la qualité globale du filtrage de la distribution de masses de la figure 3.3a est de $(0.51 + 0.65 + 0.77)/3 = 0.64$.*

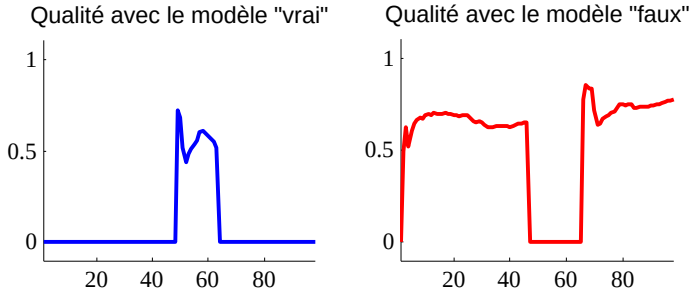



**FIG. 3.4 :** *Evolution de la qualité pour le filtrage de la figure 3.3.*

## 3.5 Estimation des paramètres du filtre

Les seuils importants utilisés dans le filtre sont :
- *le seuil d'arrêt* $\mathbb{T}_s$ utilisé pour la détection de changement,
- *le coefficient d'oubli* $\lambda$ intervenant sur le calcul de la somme cumulée,
- *les coefficients des modèles* $\gamma_{\mathcal{T}}$ et $\gamma_{\mathcal{F}}$.

Ces trois paramètres influent conjointement sur les résultats du filtre ce qui complique leur réglage. Les tailles des fenêtres temporelles intervant sur la largeur maximale de la transition ($\mathcal{W}$) et sur l'adaptation du seuil de tolérance au conflit ($N$) sont fixées par expertise. Dans un grand nombre de tests sur l'analyse de vidéos, nous avons choisi $\mathcal{W} = 3$ et $N = 5$ (ces valeurs correspondent à $3/25 \approx 0.12$ secondes et $5/25 \approx 0.2$ secondes respectivement).

### 3.5.1 Données d'apprentissage

Afin d'optimiser les paramètres, des critères doivent être définis. Dans le cadre de l'analyse de vidéos, les critères de rappel et de précision sont généralement utilisés. Leur calcul nécessite de disposer d'un ensemble d'apprentissage composée de fonctions de croyance temporelles annotées. La vérité terrain donne l'ensemble des éléments corrects $|c|$. L'algorithme de détection génère un ensemble de réponses $|r|$ que l'on compare à la vérité terrain. On note $|c \cap r|$ l'intersection des ensembles de données correctes ($|c|$) et retrouvées ($|r|$). Les définitions des critères utilisés sont les suivantes :

**Rappel :** Le rappel est donné par le rapport entre le nombre de données correctement retrouvées sur le nombre de données correctes, c'est à dire $R = \frac{|c \cap r|}{|c|}$. Le rappel est grand si l'algorithme parvient à détecter les éléments recherchés même s'ils sont noyés dans des éléments incorrects.

**Précision :** La précision est donnée par le rapport entre le nombre de données correctement retrouvées sur le nombre de données retrouvées par l'algorithme, c'est à dire $P = \frac{|c \cap r|}{|r|}$. La précision est grande si les éléments détectés sont corrects.

**Mesure $F_1$ :** Une mesure globale appelée mesure $F_1$ est utilisée comme un compromis entre rappel et précision. Elle est donnée par $F_1 = \frac{2 \cdot R \cdot P}{R+P}$.

Rappel, précision et mesure $F_1$ sont donc donnés par :

$$P = \frac{|c \cap r|}{|r|} \qquad R = \frac{|c \cap r|}{|c|} \qquad F_1 = \frac{2 \cdot R \cdot P}{R+P} \tag{3.24}$$

### 3.5.2 Seuil d'arrêt $\mathbb{T}_s$ et coefficient d'oubli $\lambda$

Pour l'estimation de la valeur du seuil d'arrêt $\mathbb{T}_s$ et du coefficient d'oubli $\lambda$, nous considérons deux procédures. La première étudie l'influence conjointe des paramètres sur les valeurs de rappel, précision et mesure $F_1$, et la seconde est une estimation plus grossière mais pouvant suffire si l'on connaît suffisamment les données traitées.

On suppose disposer d'un jeu de données d'apprentissage formées de croyances temporelles sur un état et d'une vérité terrain indiquant où l'état est vrai. Un exemple de telles données est illustré sur la figure 3.3a où la référence est donnée par le trait noir et gras sur l'axe temporel. Les croyances sont définies sur $\Omega_t = \{T_t, F_t\}$ et évoluent au cours du temps. La vérité terrain fournit en particulier l'instant de départ où l'état est vrai, on le note $t_d$.

**Etude conjointe** Le seuil d'arrêt $\mathbb{T}_s$ et le coefficient d'oubli $\lambda$ ont une influence conjointe importante sur les résultats du filtrage.

Si le seuil d'arrêt $\mathbb{T}_s$ est trop faible alors il sera rapidement atteint par la somme cumulée du conflit ce qui entrainera des changements de modèles précoces. Dans le cas où $\mathbb{T}_s$ est trop grand, il se peut qu'aucun changement de modèle n'ait lieu car le seuil sera atteint difficilement par la somme cumulée.

Quant à la valeur du coefficient d'oubli $\lambda$, elle permet de gérer l'amplitude de la somme cumulée. Si $\lambda$ est faible ($\approx 0$) alors la somme cumulée est fortement pondérée (cf Eq. 3.13) et il est donc nécessaire que $\mathbb{T}_s$ soit faible afin que le seuil soit atteint par la somme cumulée. Lorsque $\lambda$ est grand ($\approx 1$) alors $\mathbb{T}_s$ doit être relativement élevé pour être robuste aux fausses alarmes.

L'estimation d'un couple $(\mathbb{T}_s^*, \lambda^*)$ optimal pour un ensemble d'apprentissage donné (et donc pour une application donnée) consiste à tracer les valeurs de rappel, précision et/ou mesure $F_1$ en fonction des paramètres $\mathbb{T}_s$ et $\lambda$. Le couple $(\mathbb{T}_s^*, \lambda^*)$ est choisi tel qu'il maximise l'un des critères (le choix se fait en fonction de l'application). Ce type d'expérience a été menée dans [RPRP07] pour l'analyse de vidéos de sports à partir d'une base de vidéos annotées manuellement.

**Exemple 18.** *La figure 3.5 (tirée de [RRP07c]) montre l'évolution de la mesure $F_1$ pour des sauts en hauteur avec deux types de détecteur d'action (haut : modélisation heuristique basée sur les sous-ensembles flous, bas : issues de mélanges Gaussiens après apprentissage) et en fonction de quelques valeurs de ces paramètres. Cette figure montre que :*

- *il est difficile de trouver un réglage unique pour les trois actions dans le cas du premier ensemble de détecteurs (figure du haut). Un couple possible est* $(3, 0.85)$ *assurant une mesure* $F_1$ *un peu inférieure à* $70\%$ *pour les trois actions.*
- *le couple* $(2.2, 0.87)$ *assure une mesure* $F_1$ *élevée, proche ou supérieure à* $80\%$ *pour les trois actions et donc un couple unique peut être déterminé pour le deuxième ensemble de détecteurs d'action (figure du bas).*

L'étude conjointe peut prendre du temps en fonction de la taille des données à traiter et du nombre de couples. Une autre méthode, plus simple mais plus grossière, est proposée dans le paragraphe suivant.

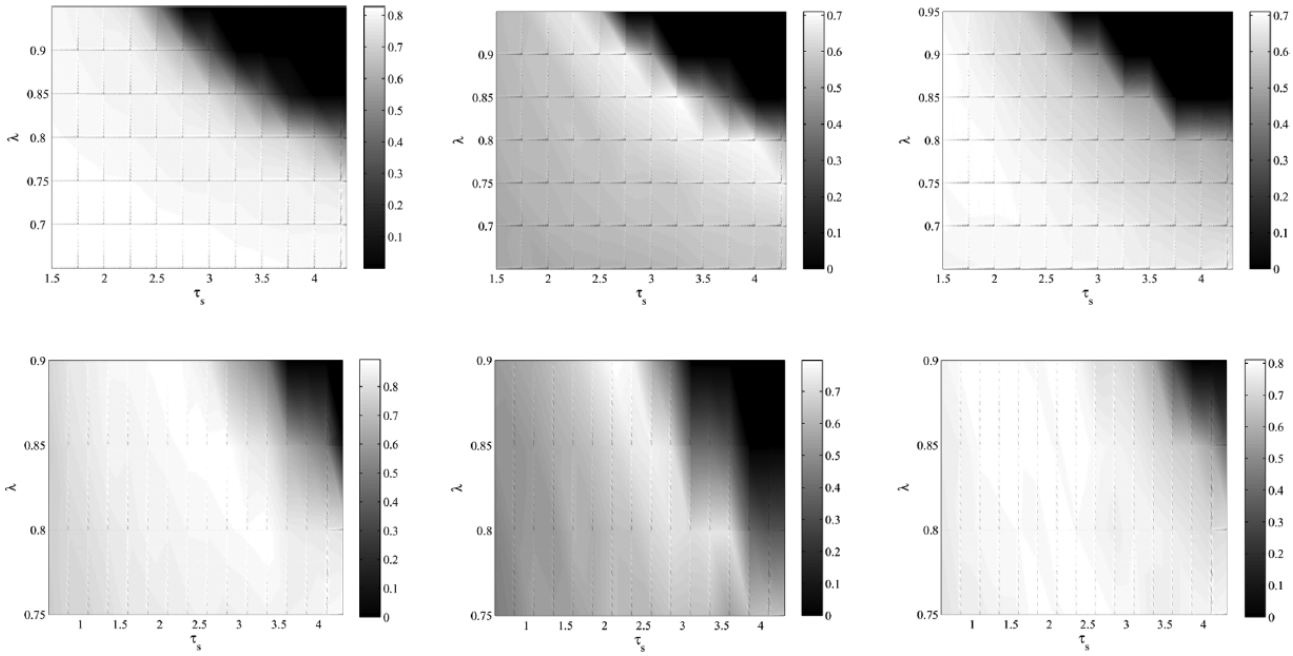



**FIG. 3.5 :** *Evolution de la mesure* $F_1$ *en fonction des couples* $(\mathbb{T}_s, \lambda)$ *pour la détection des actions* courir, sauter *et* chuter *dans un saut en hauteur (mesure moyennée sur* $10$ *sauts en hauteur). Deux ensembles de résultats sont présentés : détection des actions par la fusion de BBA issues d'une modélisation heuristique basée sur les sous-ensembles flous (haut) et issues de mélanges Gaussiens après apprentissage intensif (bas). Image tirée de [RRP07c].*

**Etude indépendante** L'étude indépendante permet une estimation plus grossière et beaucoup plus rapide que précédemment dans le cas où les données à traiter sont bien connues par l'utilisateur.

Une solution pour l'estimation d'un couple $(\mathbb{T}_s^*, \lambda^*)$ consiste en une procédure itérative débutant par le choix d'une pondération initiale $\lambda_0$ suivi d'un filtrage avec les caractéristiques suivantes :

1. initialiser le filtre avec le modèle $\mathcal{F}$ (état faux),
2. fixer le seuil d'arrêt à $\mathbb{T}_s = \infty$ afin d'interdire tout changement de modèle.

Puis appliquer le TCF. Une fois les données filtrées, on dispose d'une valeur de somme cumulée $\mathbf{CS}(t)$ pour chaque instant $t$ des données d'apprentissage. A l'instant $t_d$, correspondant à l'instant de départ de l'action donnée par la vérité terrain, la valeur de la somme cumulée (pour la valeur de $\lambda$ choisie) peut être vue comme une approximation du seuil d'arrêt, c'est à dire ${\mathbb{T}_s}^* = \mathbf{CS}(t_d)$.

Lorsque les données sont bruitées, il est possible que ce seuil ne soit pas optimal dans le sens où à d'autres instants ne correspondant pas à l'état vrai, ce seuil soit franchi. Dans ce cas, la pondération $\lambda$ de la somme cumulée peut être diminuée, par exemple $\lambda_1 = \sigma \cdot \lambda_0$, où $\sigma \in ]0, 1[$ est un taux de décroissance, et la procédure réitérée.

**Exemple 19.** *La figure 3.6 présente un exemple d'estimation pour une action* sauter *dans une vidéo de saut à la perche. La figure 3.6a illustre les BBA temporelles fournies par le détecteur d'action* sauter. *On cherche à estimer le seuil d'arrêt $\mathbb{T}_s$. On connaît pour cela la vérité terrain, indiquée par le trait noir et gras sur chacune des figures. D'après cette vérité terrain, l'action* sauter *débute en $t_d = 63$ (et se termine en $t = 103$). Comme décrit précédemment, on applique tout d'abord le filtre avec le modèle $\mathcal{F}$ et un seuil d'arrêt $\mathbb{T}_s = \infty$. Le résultat du filtrage est donné sur la figure 3.6b et celui de la somme cumulée du conflit $\mathbf{CS}(t)$ sur la figure 3.6c. Le seuil d'arrêt optimal (d'après la vérité terrain) est ${\mathbb{T}_s}^* = \mathbf{CS}(t_d) \approx 1.6$. Le problème avec ce seuil est qu'il engendre une fausse détection vers l'instant $t = 8$. On choisit d'augmenter le seuil d'arrêt pour qu'il reste insensible à cette fausse détection et on fixe $\mathbb{T}_s = 2.1$. Finalement, le filtre est appliqué sur la BBA temporelle de la figure 3.6a avec ce seuil et le résultat est donné sur la figure 3.6d.*

### 3.5.3 Coefficients des modèles $\mathcal{T}$ et $\mathcal{F}$

Pour chacun des deux modèles $\mathcal{M} \in \{\mathcal{T}, \mathcal{F}\}$, il existe un coefficient $\gamma_{\mathcal{M}}$. Ces coefficients lient la masse de croyance sur une hypothèse entre deux instants successifs, par exemple $T^{f-1}$ et $T^f$ pour le modèle $\mathcal{T}$, et peuvent être interprétés comme des croyances conditionnelles (Eq. 3.2). Pour estimer la valeur des coefficients $\gamma_{\mathcal{T}}$ lorsque l'état est vrai, nous proposons la procédure suivante. Notons que cette procédure est facilement généralisable pour estimer $\gamma_{\mathcal{F}}$ dans le cas où l'état est faux.

On suppose que l'on dispose d'un jeu de $n$ BBA $[\tilde{m}^{\Omega^1}; \tilde{m}^{\Omega^2}; \ldots \tilde{m}^{\Omega_t}; \ldots \tilde{m}^{\Omega^n}]$ traduisant la véracité d'un état à des instants $t$ et définies sur un espace binaire. Ces échantillons, correspondant à des instants où la vérité indique que l'état est vrai, sont classiquement fournis par un système de fusion. Nous utilisons ces échantillons pour estimer la *croyance conditionnelle moyenne* de l'hypothèse « l'état est vrai » (c'est à dire $T_t$) :

$$\gamma_{\mathcal{T}} = \tilde{m}^{\Omega_t}_{\text{moy}}[T_{t-1}](T_t) \tag{3.25}$$

Pour cela, nous calculons tout d'abord la BBA de chaque couple $(m^{\Omega_{t-1}}, m^{\Omega_t})$. Cette BBA est obtenue en étendant les deux BBA du couple sur l'espace commun donné par $\Omega_{t-1} \times \Omega_t$ puis en les combinant conjonctivement. La BBA obtenue est ensuite conditionnée sur la vérité terrain à $t-1$, c'est à dire « l'état est vrai », puis marginalisée sur l'espace courant $\Omega_t$. On obtient donc :

$$\tilde{m}^{\Omega_t}_{\text{moy}}[T_{t-1}] = \left( \frac{1}{n-1} \cdot \sum_{t=2}^{n} \left( \tilde{m}^{\Omega_{t-1} \uparrow \Omega_{t-1} \times \Omega_t} \textcircled{\cap} \tilde{m}^{\Omega_t \uparrow \Omega_{t-1} \times \Omega_t} \right) \right) [T_{t-1}]^{\downarrow \Omega_t} \tag{3.26}$$

Le conflit dans les données d'apprentissage peut générer du conflit dans l'estimation et mener à $\tilde{m}^{\Omega_t}_{\text{moy}}[T_{t-1}](\emptyset_t) \geq 0$. Nous choisissons de normaliser la BBA résultante en utilisant l'équation 2.3 menant à :

$$\gamma_{\mathcal{T}} = \frac{\tilde{m}^{\Omega_t}_{\text{moy}}[T_{t-1}](T_t)}{1 - \tilde{m}^{\Omega_t}_{\text{moy}}[T_{t-1}](\emptyset_t)} \tag{3.27}$$

Pour obtenir $\gamma_{\mathcal{F}}$, il suffit d'appliquer le même schéma mais sur des BBA où la vérité terrain indique que l'état est *faux* et en conditionnant par $F_{t-1}$ dans l'équation 3.26. On obtient alors l'estimée de $\gamma_{\mathcal{F}}$

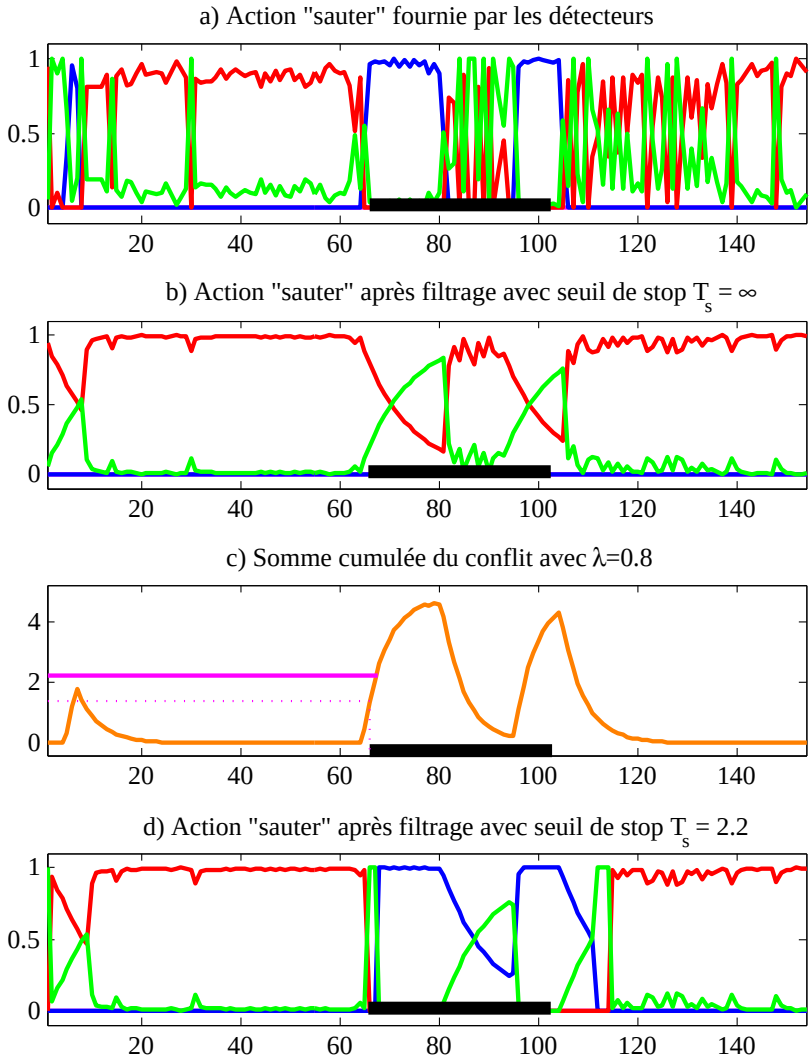



**FIG. 3.6 :** *Estimation du seuil d'arrêt : a) données d'apprentissage avec la vérité terrain indiquée par un trait noir, b) le filtrage avec un seuil d'arrêt infini, c) évolution de la somme cumulée du conflit lorsque le seuil d'arrêt est infini et d) le filtrage avec un seuil d'arrêt fixé à 2.1 obtenu comme indiqué à l'exemple 19.*

par :

$$\gamma_{\mathcal{F}} = \frac{\tilde{m}^{\Omega_t}_{\text{moy}}[F_{t-1}](F_t)}{1 - \tilde{m}^{\Omega_t}_{\text{moy}}[F_{t-1}](\emptyset_t)} \tag{3.28}$$

**Exemple 20.** *Nous utilisons les données d'apprentissage de la figure 3.7 auxquelles nous appliquons la procédure d'estimation des coefficients des modèles $\gamma_{\mathcal{T}}$ et $\gamma_{\mathcal{F}}$. L'intervalle $[12, 71]$ est utilisé pour le coefficient $\gamma_{\mathcal{T}}$ (Eq. 3.26) tandis que l'intervalle $[74, 193]$ est utilisé pour estimer $\gamma_{\mathcal{F}}$ (Eq. 3.26 mais avec conditionnement sur $F_{t-1}$). On obtient alors les deux BBAs $m^{\Omega_t}_{moy}[T_{t-1}]$ et $m^{\Omega_t}_{moy}[F_{t-1}]$ suivantes :*

$$\begin{array}{rclrcl}
\boldsymbol{m^{\Omega_t}_{moy}[T_{t-1}](T_t)} & = & \mathbf{0.755} & m^{\Omega_t}_{moy}[F_{t-1}](T_t) & = & 0.024 \\
m^{\Omega_t}_{moy}[T_{t-1}](F_t) & = & 0.052 & \boldsymbol{m^{\Omega_t}_{moy}[F_{t-1}](F_t)} & = & \mathbf{0.917} \\
m^{\Omega_t}_{moy}[T_{t-1}](T_t \cup F_t) & = & 0.193 & m^{\Omega_t}_{moy}[F_{t-1}](T_t \cup F_t) & = & 0.059
\end{array} \tag{3.29}$$

*Les coefficients des modèles sont donc :*

$$\gamma_{\mathcal{T}} = 0.755 \qquad \gamma_{\mathcal{F}} = 0.917 \tag{3.30}$$

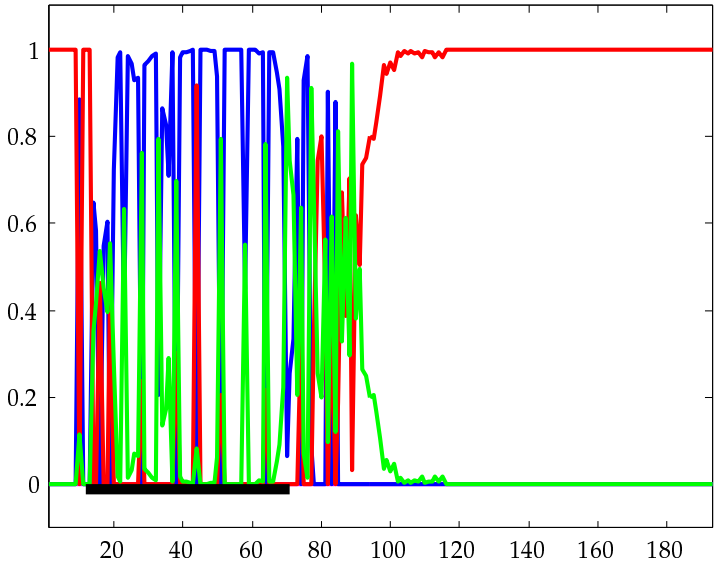



**FIG. 3.7 :** *Un jeu de données d'apprentissage : action « course » dans un saut en hauteur. La vérité terrain est donnée par le trait noir sur l'axe temporel.*

### 3.5.4 Initialisation du filtre

Le filtre temporel crédibiliste fonctionne en ligne (la BBA à $t$ est calculée à partir des informations à $t-1$). Lors de l'initialisation, le modèle qui correspond le mieux aux premières mesures est sélectionné. Pour cela, le calcul de la CUSUM est réalisé sur les premièrs instants pour les deux modèles puis celui présentant la plus petite CUSUM est choisi.

## 3.6 Amélioration du filtre

Le filtre que nous avons proposé impose un choix *binaire* et *strict* du modèle d'évolution à chaque instant (soit $\mathcal{T}$ ou soit $\mathcal{F}$ est appliqué). Il pourrait être intéressant de lever ces deux contraintes et d'étudier les aspects suivants :

- **La commutation n'est plus basée sur un seuillage –** Le réglage d'un seuil peut être critique. Une solution consiste à considérer une description floue du seuil voire une fonction de croyance sur les modèles. Dans ce cas, nous tendrons vers une généralisation des modèles à sauts [PRCM99] (§ 3.1).
- **La fonction de croyance temporelle n'est plus binaire –** Comment appliquer le filtre lorsque la fonction de croyance à filtrer est définie sur un cadre de discernement $\Omega$ de cardinalité $n = |\Omega| > 2$ ? Nous proposons une approche basée sur un grossissement du cadre de discernement.

Nous présentons ci-dessous des pistes d'amélioration possibles et soulevons les problèmes inhérents à ces propositions.

### 3.6.1 Changement de modèle crédibiliste

Le filtrage avec le filtre temporel crédibiliste (TCF) peut être performant mais il est nécessaire de bien régler le seuil d'arrêt. Afin de faciliter son réglage, une solution consiste à décrire la commutation par une BBA. De cette manière, à chaque instant $t$, une BBA est générée qui quantifie la croyance sur les modèles devant être utilisés à $t$. On parle alors de modèle à sauts crédibilistes. La détermination du meilleur état à chaque instant est complexe car il est conditionné par le modèle. Pour une distribution temporelle de durée $T$, il existe $2^T$ séquences de modèles possibles (deux modèles sont possibles à chaque instant) et après avoir déterminé la meilleure séquence de modèles, il est encore nécessaire de déterminer la meilleure séquence d'états. Dans le cadre des modèles à sauts probabilistes [PRCM99], une solution basée sur un algorithme de Viterbi approximatif et un filtre de Kalman a été proposée.

Dans le cadre du TCF avec modèle à sauts crédibilistes, un problème lié au conflit peut apparaître. En effet, la consonance de la BBA à chaque instant n'est plus assurée en sortie du filtre puisque le modèle n'est plus connu avec certitude et donc la prédiction et la mise à jour peuvent induire du conflit qu'il apparaît difficile de redistribuer. Ce conflit pourrait par ailleurs être utilisé pour déterminer le meilleur modèle à chaque instant.

### 3.6.2 Généralisation du filtre à N états

En fonction de l'application, il peut être intéressant de filtrer une fonction de croyance définie sur un cadre de discernement non binaire et de cardinalité $N = |\Omega| > 2$. Nous proposons une méthode permettant d'exploiter le filtre TCF. Pour cela, on pose $\Omega_t^s = \{E_t^1, E_t^2 \dots E_t^N\}$ le cadre de discernement de la fonction de croyance à filtrer et on dispose à chaque instant d'une BBA temporelle $m^{\Omega_t^s}$ qu'il s'agit de filtrer. Les hypothèses $E_t^i$ seront appelées *états* dans la suite.

Pour réaliser le filtrage, nous proposons de se ramener à un cadre de discernement binaire à l'aide des opérations sur les cadres. La proposition est représentée sur la figure 3.8 : nous effectuons $N$ grossissements du cadre de discernement $\Omega_t^s$, c'est à dire autant de grossissement que d'hypothèses, où chaque grossissement est binaire sur $\Omega_t^i = \{T_t^i, F_t^i\}$ signifiant que l'hypothèse $E_t^i \in \Omega_t^s$ est vraie ou fausse :

$$\begin{array}{rcl} \Omega_t^i & = & \{T_t^i, F_t^i\} \\ T_t^i & : & \text{l'hypothèse } E_t^i \text{ est vraie} \\ F_t^i & : & \text{l'hypothèse } E_t^i \text{ est fausse} \end{array} \tag{3.31}$$

Ainsi, il est possible d'appliquer le filtre TCF puisque nous nous sommes ramenés à des BBA temporelles binaires. La procédure proposée consiste en $4$ étapes illustrés sur la figure 3.8 :

1. créer $N$ BBA binaires définies sur à partir de la BBA initiale,
2. filtrer les $N$ BBA binaires indépendamment avec le filtre TCF,
3. étendre les $N$ BBA filtrées avec l'opérateur d'extension vide sur le cadre initial $\Omega_t^s$,
4. combiner les $N$ BBA étendues avec la CRC.

**Etape 1 : grossissement** L'espace $\Omega_t^i$ est un grossissement de $\Omega_t^s$ et donc nous devons définir l'application $\rho : 2^{\Omega_t^s} \rightarrow 2^{\Omega_t^i}$ afin de transférer les masses des différentes propositions de $\Omega_t^s$ vers les propositions dans $2^{\Omega_t^i} = \{\emptyset_t^i, \{T_t^i\}, \{F_t^i\}, \{T_t^i, F_t^i\}\}$ :

$$\begin{array}{rcl} \rho : 2^{\Omega_t^s} & \rightarrow & 2^{\Omega_t^i} \\ m^{\Omega_t^s}(B) & \mapsto & m^{\Omega_t^i}(\rho(B)) \end{array} \tag{3.32}$$

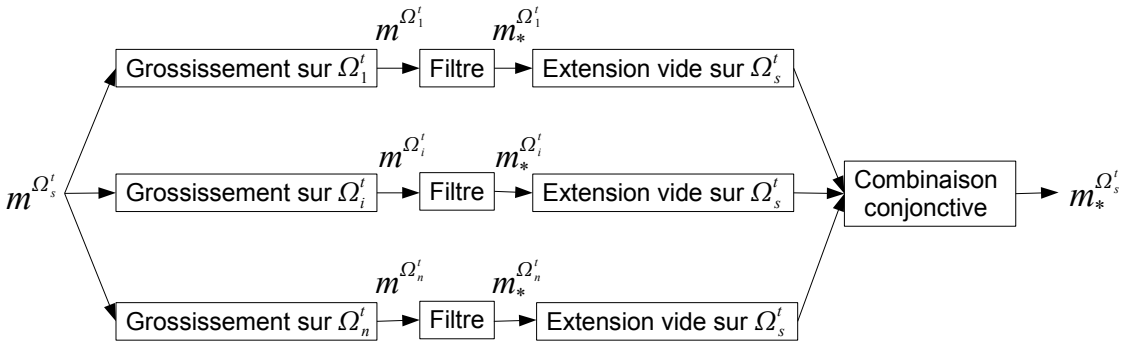



**FIG. 3.8 :** *Application du TCF pour le filtrage de fonctions de croyance non binaires.*

Le grossissement est illustré sur la figure 3.9 où l'on cherche à filtrer la masse sur un élément $E_t^i$. Cet élément est donc associé à $T_t^i$ et les autres éléments, c'est à dire $\overline{E}_t^i$, sont associés à $F_t^i$.

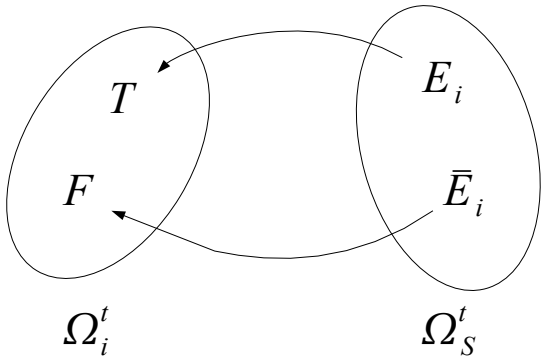



**FIG. 3.9 :** *Illustrations du grossissement pour le filtrage de fonctions de croyance définies sur des espaces non binaires.*

Une application $\rho$ assurant une consistance des éléments focaux dans $2^{\Omega_t^i}$ est la suivante :

- La masse $m^{\Omega_t^i}(T_t^i)$ correspond à la part de croyance sur le fait que l'état est vrai. Le PMI indique que seule la masse $m^{\Omega_t^s}(E_t^i)$ correspond effectivement à cette connaissance.
- La masse $m^{\Omega_t^i}(F_t^i)$ correspond à la part de croyance sur le fait que l'état est faux. Le PMI indique que toutes les masses dont les éléments focaux ne contiennent pas $E_t^i$ correspondent à cette connaissance.
- La masse $m^{\Omega_t^i}(\Omega_t^i)$ correspond à la part de croyance sur le fait que l'état est vrai ou faux. Le PMI indique que toutes les masses contenant $E_t^i$ et dont les éléments focaux sont de cardinalité supérieure à l'unité correspondent à cette connaissance.
- La masse $m^{\Omega_t^i}(\emptyset_t^i)$ a seulement $m^{\Omega_t^s}(\emptyset_t)$ pour correspondance.

La BBA binaire $m^{\Omega_t^i}$ de l'état $E_t^i$ est donc obtenue de la manière suivante :

$$\begin{array}{lcl} m^{\Omega_t^i}(T_t^i) & \leftarrow & m^{\Omega_t^s}(E_t^i) \\ m^{\Omega_t^i}(F_t^i) & \leftarrow & \displaystyle\sum_{\substack{E_t^i \cap B^i = \emptyset \\ B^i \neq \emptyset}} m^{\Omega_t^s}(B^i) \\ m^{\Omega_t^i}(T_t^i \cup F_t^i) & \leftarrow & \displaystyle\sum_{E_t^i \subset B^i} m^{\Omega_t^s}(B^i) \\ m^{\Omega_t^i}(\emptyset_t) & \leftarrow & m^{\Omega_t^s}(\emptyset_t) \end{array} \tag{3.33}$$

**Etape 2 : filtrage** On dispose de $n$ BBA, une pour chaque état $E_t^i$ et chacune définie sur un espace binaire. Elles peuvent être traitées par le filtre TCF présenté précédemment. Les $n$ BBA filtrées sont $m_*^{\Omega_t^i}$.

**Etape 3 : raffinement** Après filtrage, il est possible d'obtenir une BBA sur l'espace $\Omega_t^s$ à partir des BBA sur chaque état. Pour cela, on suit le même raisonnement que sur la figure 3.9 et on calcule la BBA sur l'espace raffiné $\Omega_t^s$ à chaque instant de la manière suivante :

$$\begin{array}{rcl} m_i^{\Omega_t^s}(E_t^i) & \leftarrow & m^{\Omega_t^i}(T_t^i) \\ m_i^{\Omega_t^s}(\Omega_t^s) & \leftarrow & m^{\Omega_t^i}(T_t^i \cup F_t^i) \\ m_i^{\Omega_t^s}(\Omega_t^s \backslash E_t^i) & \leftarrow & m^{\Omega_t^i}(F_t^i) \end{array} \tag{3.34}$$

**Remarque 3.1.** *La masse $m^{\Omega_t^i}(\emptyset_t)$ n'est pas utilisée car le TCF assure une sortie sans conflit. De plus, la sortie du TCF étant consonante, ceci implique que soit $E_t^i$ ou soit $\Omega_t^s \backslash E_t^i$ est un élément focal à un instant donné. Par conséquent, les $n$ BBA obtenues sont des BBA simples.*

**Etape 4 : combinaison** Disposant des $n$ BBA définies sur $\Omega_t^s$, il est possible de les combiner afin d'établir la croyance sur $\Omega_t^s$ prenant en compte les différents filtrages. Cette BBA est obtenue par la combinaison conjonctive des BBA fournies par les $n$ filtres :

$$m^{\Omega_t^s} = \bigcirc\!\!\!\!\cap_{i \in \{1 \ldots N\}} m_i^{\Omega_t^s} \tag{3.35}$$

**Quelques remarques** Une première remarque sur la méthode concerne le fait que l'on dégrade la BBA initiale. En effet, l'opération de grossissement (Eq. 3.34) n'est généralement pas réversible. On peut même dire que l'on perd de l'information car la granularité du cadre après grossissement est plus faible.

De plus, les BBAs après raffinements sont des BBA simples et cette propriété provient des redistributions proposées dans la définition du filtre. Le problème est que les BBA simples sont combinées avec un opérateur conjonctif et par conséquent du conflit risque d'apparaître.

## 3.7 Conclusion

Le filtre que nous avons proposé permet d'une part d'estimer la BBA sur la véracité d'un état en fonction des observations faites sur le système et de l'état à l'instant précédent. D'autre part, ce filtre permet de filtrer les BBA c'est à dire d'obtenir des BBA moins sensibles aux défauts de perception. Les BBA obtenues sont consonantes et sans conflit.

Le filtre repose sur le principe « prédiction – mise à jour » classiquement utilisé dans les méthodes de filtrage. Notre apport tient essentiellement dans l'aspect *commutation de modèles* entre les modèles *l'état est vrai* et *l'état est faux* et l'aspect *exploitation du conflit*. La commutation de modèles est inspirée de méthodes développées dans la théorie des probabilités et elle est gérée par un mécanisme de seuillage du conflit apparaissant lors de la phase de « mise à jour » et plus précisément lors de la fusion entre la prédiction et la mesure des capteurs. L'application du filtre nécessite l'utilisation de trois paramètres importants pour lesquels une procédure d'estimation a été proposée.

Le filtre exploite la relation entre les BBA d'un même état à deux instants consécutifs. Il s'agit à présent de tirer parti de la relation entre les états. C'est ce qui est proposé dans les deux chapitres suivants où nous traitons le problème de la reconnaissance de séquences d'états dans le cadre des fonctions de croyance.

# Chapitre 4

# Reconnaissance déterministe de séquences d'états par machine à états finis crédibilistes

## 4.1 Introduction

Dans le cadre de l'analyse de systèmes dynamiques, les croyances fournies par les sources observatrices traduisent généralement une opinion sur un *état* pris par le système. Dans les applications réelles, les états sont multiples et le système *évolue* d'un état à l'autre au cours du temps suivant des *séquences* particulières. L'aspect *dynamique* intervient à deux niveaux :

- Sur un même état et entre deux instants consécutifs : le filtre TCF proposé dans le chapitre précédent tire parti de cette information.
- Sur la séquence des états au cours du temps : cette information permet de prendre en compte les relations entre les états.

Une méthode de reconnaissance de séquences doit, en plus des informations de causalité entre les états de la séquence, intégrer les informations temporelles entre deux instants successifs.

Deux grandes familles de méthodes de reconnaissance de séquences existent : les méthodes IA (*Intelligence Artificielle*), pour lesquelles les modèles de séquences sont bien connus et fournis par des experts du domaine d'application, et les méthodes statistiques où les modèles de séquences sont inconnus et déterminés à partir d'un processus d'apprentissage. Les méthodes statistiques font l'objet du chapitre suivant. Dans ce chapitre, nous nous intéressons aux méthodes IA dans le cadre des fonctions de croyance.

Peu de travaux sur la reconnaissance de séquences d'états dans le cadre des fonctions de croyance ont été réalisés. On peut citer deux méthodes proposées par Rombaut et al. : la première est un *réseau de Pétri crédibiliste* (BPN : *Belief Petri Net*) élaboré lors de la thèse de I. Jarkass [RJD99] et la seconde est le système IDRES (*Intelligent Driving Recognition with Expert System*) proposé avec J.M. Nigro [NR03]. Ces deux systèmes ont été appliqués à la reconnaissance de manoeuvres d'un conducteur dans le cadre de l'assistance à la conduite automobile et ont été comparés dans [RLRNJ00]. Dans le paragraphe ci-dessous, nous décrivons les points clés du BPN qui représente une méthode générale proche de l'ordonnanceur CSS alors que IDRES est un système expert dédié à une application particulière.

Notons que les réseaux évidentiels (§ 2.2.1.3) permettraient de décrire des séquences d'états sous la forme de graphes orientés. Ils ont été appliqués à des problèmes de diagnostic mais n'ont pas été appliqués à la reconnaissance de séquences et l'aspect temporel n'a jamais été explicité dans ces

réseaux. Nous proposerons dans le chapitre suivant une méthode pour traiter des fonctions de croyance temporelles utilisant des outils initialement développés dans le cadre des réseaux évidentiels.

**Le réseau de Pétri crédibiliste** Le BPN décrit une séquence sous la forme d'un graphe orienté. Le graphe est composé de $P$ *places* (équivalentes à des nœuds) et dans le modèle proposé dans [RJD99], chaque place est reliée au plus à une place amont et au plus une place avale.

A la différence de ce que nous proposons, il s'agit de calculer, à un instant $t$ donné, la fonction de croyance sur l'ensemble des places $m^{\Omega^t}$ avec $\Omega^t = \{P_1^t, P_2^t \dots P_K^t\}$, au lieu de $K$ croyances sur chacune des places, à partir de :

- la fonction de croyance sur l'ensemble des places à l'instant précédent $m^{\Omega^{t-1}}$. A l'initialisation, une distribution a priori $m^{\Omega^0}$ doit être choisie. Cette BBA peut être vide ou prendre une forme quelconque (cf. tableau 2.1).
- l'ensemble des fonctions de croyance sur les transitions entre deux places consécutives $P_k$ et $P_{k+1}$. Chacune de ces BBA est définie sur $\Theta_k^t = \{T_k^t, F_k^t\}$ et représente la croyance sur le fait que la transition de $P_k$ vers $P_{k+1}$ est vraie ($T_k$) ou fausse ($F_k$).

La croyance à un instant $t$ sur l'ensemble des places est déterminée par application du GBT (Eq. 2.28) :

$$m^{\Omega^t}(C) = \sum_{B \subseteq \Omega^{t-1}} m^{\Omega^t}[B](C) \cdot m^{\Omega^{t-1}}(B), \forall C \subseteq \Omega^t \tag{4.1}$$

où les masses conditionnelles $m^{\Omega^t}[B]$ représentent la croyance sur les places à $t$ conditionnellement aux croyances sur les places à l'instant précédent $t-1$ et sont quantifiées par les croyances sur les transitions. Chaque transition est vue comme une source d'information donnant la croyance conditionnelle que la place avale soit vraie ou fausse conditionnellement à la place amont :

$$\begin{aligned} m^{\Omega^t}[P_k^{t-1}](\{P_{k+1}^t\}) &= m^{\Theta_k^t}(\{T_k^t\}) \\ m^{\Omega^t}[P_k^{t-1}](\{P_k^t\}) &= m^{\Theta_k^t}(\{F_k^t\}) \\ m^{\Omega^t}[P_k^{t-1}](\{P_k^t, P_{k+1}^t\}) &= m^{\Theta_k^t}(\{T_k^t, F_k^t\}) \end{aligned} \tag{4.2}$$

Le BPN peut être utilisé pour décrire des sauts d'athlétisme comme le saut en hauteur : « courir → sauter → chuter → se relever », où chaque place représente une action, par exemple *courir* est la première place. Le problème est que la règle d'évolution de la BBA correspond au « passage au plus tôt » contrairement à l'algorithme que nous proposons dans les paragraphes suivants. Cet inconvénient implique que le BPN n'est pas robuste au bruit et aux fausses alarmes. La figure 4.1 illustre le problème de robustesse du BPN où les ellipses représentent la distribution de masses : partant d'une BBA vide (réseau 1, à gauche), les transitions 1, 2 puis 3 deviennent vraies ce qui a pour effet de transférer la masse vers les places inférieures (réseaux 2 à 4, de gauche à droite). Si les transitions sont des fausses alarmes alors la masse qui est « descendue » dans le réseau sera « perdue ». L'annexe 3 illustre le problème de robustesse sur un exemple.

Dans les sections suivantes, nous décrivons successivement le principe et les mécanismes de la machine à états finis que nous proposons.

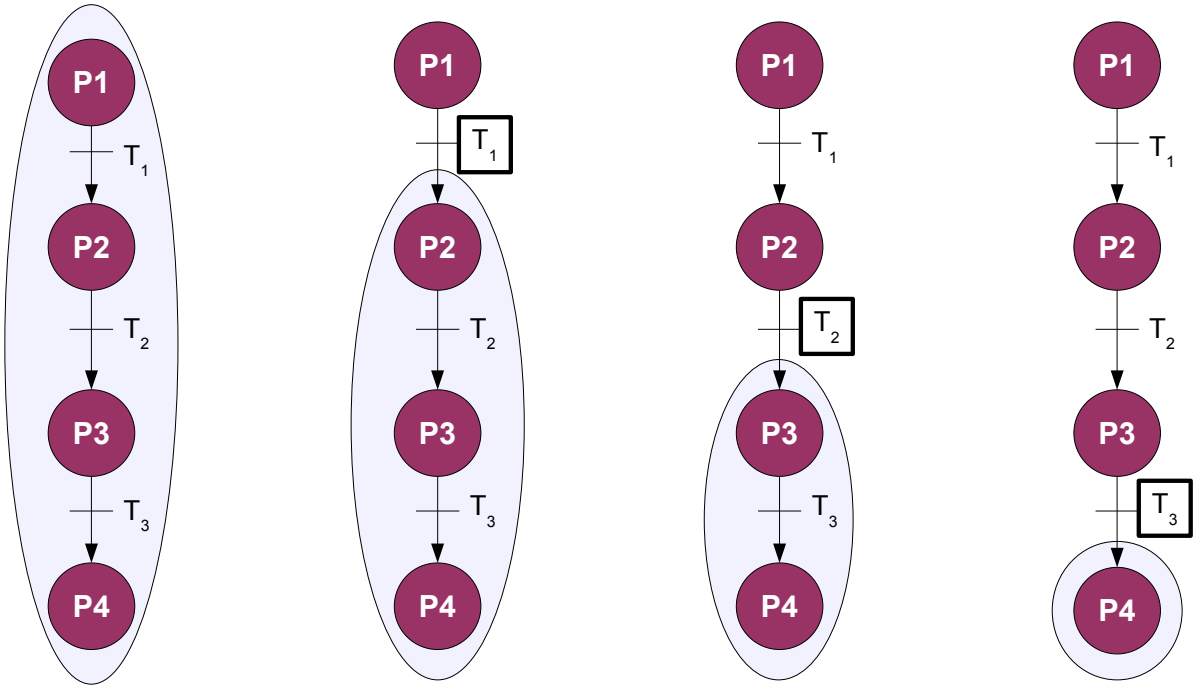



**FIG. 4.1 :** *BPN : évolution de la masse et problème des fausses alarmes.*

## 4.2 Approche crédibiliste pour la reconnaissance de séquences

On cherche à reconnaître des séquences d'états où chaque état est décrit par une fonction de croyance binaire quantifiant sa véracité. Pour cela, nous proposons une *machine à états* que nous appelons *ordonnanceur*[1] *d'états crédibilistes* (CSS : *Credal State Scheduler*) et permet de déterminer :

- le meilleur état à chaque instant,
- la meilleure séquence à chaque instant.

Le dernier point permet de classer des séquences en fonction de leur évolution. La reconnaissance « en ligne » des états et des séquences est importante pour de nombreux problèmes comme la surveillance ou la reconnaissance de formes. Le CSS est une méthode *déterministe* de reconnaissance de séquences d'états ce qui signifie qu'à partir de l'état précédent et des informations courantes, l'état en cours est connu.

### 4.2.1 Principe de l'ordonnanceur

Une séquence $S_n$ est composée de $K$ états $E_k$ successifs comme illustré sur la figure 4.2. Un état est relié au plus à un état aval et au plus un état amont. Dans une séquence, chaque état $E_k$ est décrit par une fonction de croyance temporelle binaire définie à chaque instant $t$ sur le cadre de discernement $\Omega_k^t = \{T_k^t, F_k^t\}$ et reflétant la croyance sur la véracité de l'état à $t$.

Le CSS exploite le filtre TCF décrit dans le chapitre précédent comme illustré sur la figure 4.3 : un filtre est appliqué sur chaque état (sens horizontal) et le changement de modèle est *contraint* par la séquence (sens vertical). La transition d'un état $E_k$ à l'état suivant $E_{k+1}$ entre deux instants consécutifs est déterminée par la valeur des critères de changement de modèle (§ 3.3.4) de chaque filtre.

[1] « Ordonnancer » signifie « ordonner » ou « suivre une séquence ». La méthode proposée est basé sur des concepts utilisés en informatique industrielle [HA96, KL98, MSQ03, HC05] pour l'ordonnancement de tâches.

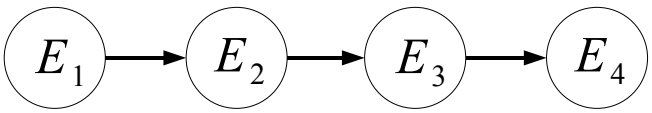



**FIG. 4.2 :** *Une séquence $S_1$ composée de 4 états.*

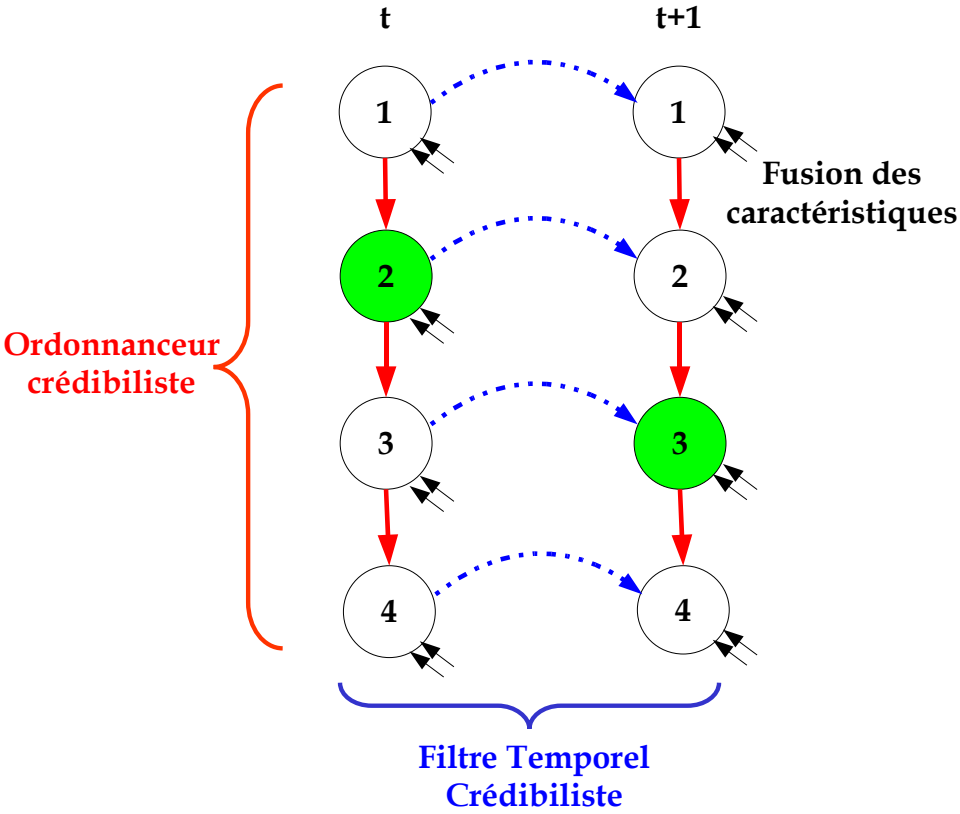



**FIG. 4.3 :** *Représentation d'une séquence de 4 états entre deux instants successifs. Le filtre TCF est utilisé pour assurer une cohérence temporelle sur chaque état (sens horizontal) et la machine à états intègre des informations sur la causalité entre les états (sens vertical).*

### 4.2.2 Définition des concepts

**Etat en cours** Un état est dit *en cours* si le filtre utilisé pour cet état utilise le modèle d'évolution « l'état est vrai $\mathcal{T}$ » lors de la prédiction. Pour une séquence donnée et à chaque instant, nous supposons qu'un seul état peut être vrai. Par conséquent, un seul modèle « l'état est vrai $\mathcal{T}$ » est disponible pour une même séquence à chaque instant et les $K-1$ autres états utilisent le modèle « l'état est faux $\mathcal{F}$ » pour la prédiction. Ainsi, l'état en cours est connu et il est décrit par une distribution de croyances simple telle que $m^{\Omega_k^t}(T_k^t) \geq 0$ et $m^{\Omega_k^t}(\{T_k^t, F_k^t\}) \geq 0$, les autres états étant décrit par une BBA simple telle que $m^{\Omega_i^t}(F_i^t) \geq 0$ et $m^{\Omega_i^t}(\{T_i^t, F_i^t\}) \geq 0$ pour $i \neq k$.

Le modèle « l'état est vrai $\mathcal{T}$ » peut être vu comme une *ressource* à laquelle les états veulent accéder tandis que les états peuvent être perçus comme des *processus* souhaitant accéder à la ressource.

**Modèle naturel et modèle contraint** Lorsque le modèle utilisé pour la prédiction sur un état est imposé par la topologie de la séquence, on dit que le modèle est *contraint* sinon, le modèle est dit *naturel*. Par exemple, un modèle est contraint lorsque deux états veulent accéder au modèle $\mathcal{T}$ à un

même instant : un des deux états pourra y accéder et le filtre associé utilisera un modèle de prédiction *naturel* tandis que le filtre associé au second état se verra *contraint* d'utiliser le modèle $\mathcal{F}$.

**Exemple 21.** *L'objectif est de reconnaître une activité « saut en hauteur » à partir d'une vidéo. Nous savons par expertise décrire cette activité humaine comme une séquence de 4 états : « courir → sauter → chuter → se relever ». Cette activité est illustrée sur la figure 4.4. Chaque état représente une action humaine et une activité est donc une séquence d'actions.*

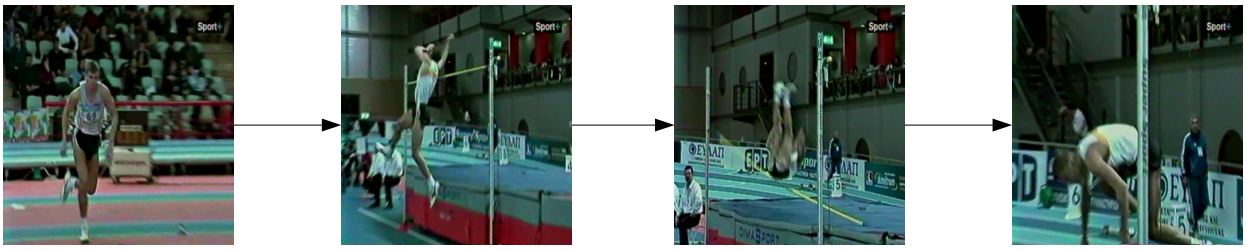



**FIG. 4.4 :** *L'activité* saut en hauteur *décrite comme une séquence de* 4 *actions.*

*On dispose de 4 sources d'informations, chacune quantifiant la véracité d'une action par une fonction de croyance binaire. La figure 4.5 illustre l'évolution des BBA temporelles sur les 4 actions fournies par ces sources.*

*La figure 4.6 illustre les BBA temporelles après filtrage par le filtre TCF. Chaque action est filtrée indépendamment des autres par un filtre qui lui est propre mais dont le réglage est similaire pour toutes les actions. Cette figure montre clairement que les actions ne sont pas bien synchronisées : par exemple les actions* courir *et* sauter *se chevauchent (image* $42$ *à* $57$*) tandis que les actions* chuter *et* se relever *sont légèrement disjointes (images* $76$ *à* $80$*). De plus, on remarque que l'action* chuter *est détectée avant le* saut *ce qui est physiquement improbable.*

*La figure 4.7 illustre les BBA temporelles après traitement par l'ordonnanceur CSS où l'on voit qu'il synchronise les actions : en sortie, les croyances sur les actions sont consonantes et si une action est vraie alors les autres actions sont fausses par définition de l'activité. La figure 4.7 permet d'illustrer les principaux concepts :*

**Etat en cours :** *sur l'intervalle* $[2, 42]$ *l'action* courir *est vraie, elle est donc l'action* en cours *(ou de façon équivalente l'état en cours). De la même manière, sur les intervalles* $[43, 59]$, $[61, 78]$ *et* $[80, 98]$ *les actions* sauter, chuter *et* se relever *sont respectivement les actions en cours.*

**Modèle contraint :** *sur l'intervalle* $[45, 58]$, *le filtre TCF sur l'action* courir *donne cette action comme étant vraie (Fig. 4.6) et le CSS contraint ce filtre à utiliser le modèle « $\mathcal{F}$ : l'action est fausse » lors de la prédiction (Fig. 4.7) car l'action en cours n'est pas l'action* courir *mais l'action* sauter. *L'action* sauter *est quant à elle filtrée avec un modèle de prédiction « $\mathcal{T}$ : l'action est vraie » sur cet intervalle et le CSS impose que ce soit la seule action vraie puisqu'une seule action (ou état) peut être vraie à un instant donné. Pour l'action* sauter, *le modèle est naturel sur ce même intervalle car il correspond aux mesures fournies par les sources et donc n'est pas contraint.*

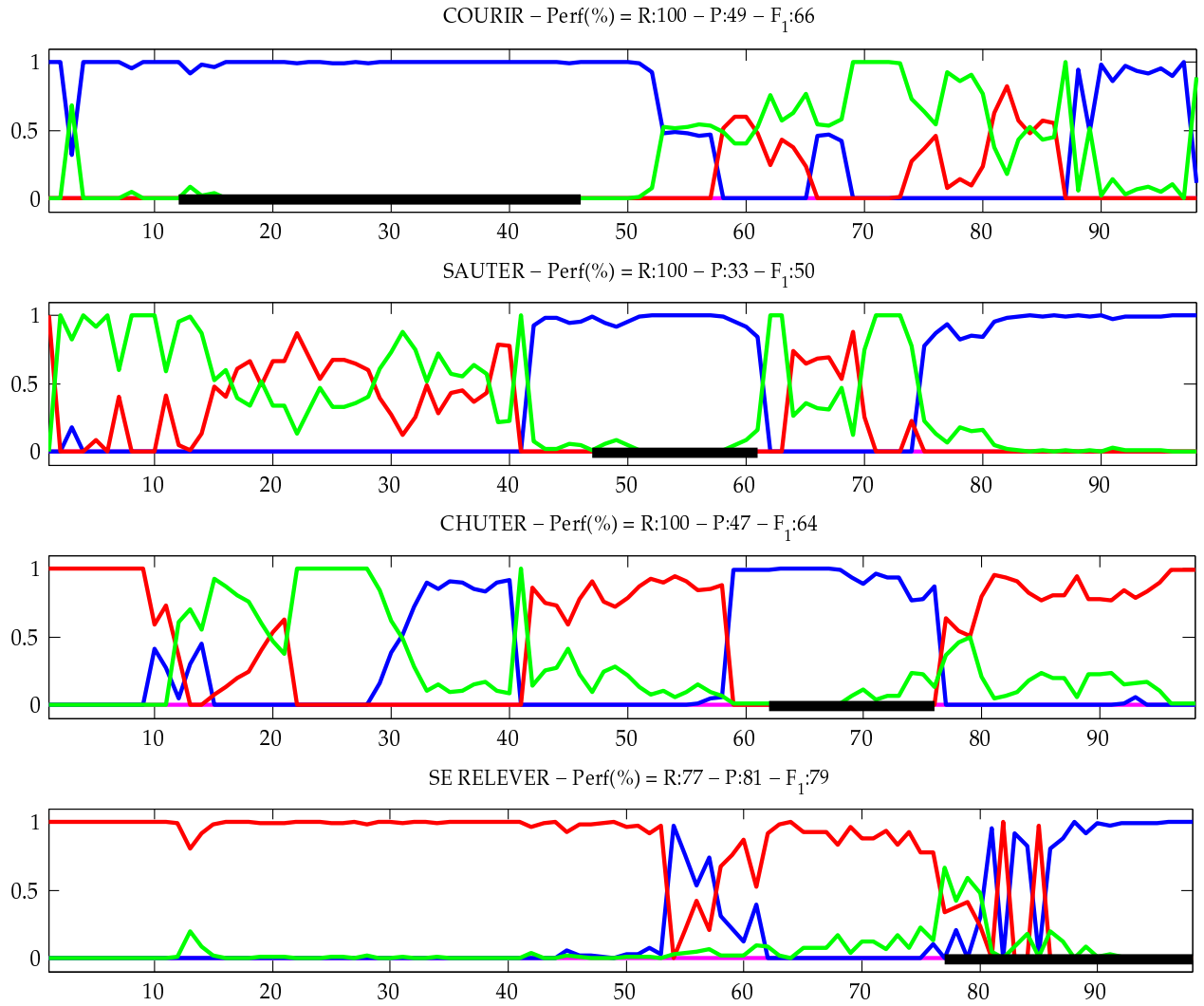



**FIG. 4.5 :** *Croyances sur les actions avant filtrage, fournies par les sources. Courbe bleue : action vraie, courbe rouge : action fausse, courbe verte : action vraie ou fausse.*

*Lorsqu'un modèle est contraint à être faux alors que l'état naturel est vrai, on remarque que la distribution de croyances sur l'action correspondante tend vers la BBA vide signifiant que toute la masse est peu à peu transférée vers le doute total* $\{T_k^t, F_k^t\}$*. Ce comportement du filtre est en accord avec le fait que les mesures et le modèle sont fortement contradictoires mais le modèle est maintenu par l'ordonnanceur à cause de la structure du graphe.*

*De la même manière, sur l'intervalle* $[33, 40]$*, le filtre sur l'action* chuter *est contraint à utiliser le modèle « $\mathcal{F}$ : l'action est fausse » (Fig. 4.7) alors que sur l'intervalle* $[61, 78]$ *(Fig. 4.6), le modèle est naturellement « $\mathcal{T}$ : l'action est vraie ».*

*Le tableau 4.1 compare les performances en terme de rappel* ($R$)*, précision* ($P$) *et mesure* $F_1$ *de :*

- *la détection des actions par les sources (ligne 1),*
- *la détection des actions après traitement par le filtre TCF (ligne 2),*
- *la détection des actions après traitement par l'ordonnanceur CSS (ligne 3).*

*On remarque l'amélioration des performances après application du filtre TCF et de manière encore plus significative après application du CSS. Généralement, le TCF et le CSS permettent d'améliorer la précision de la détection en supprimant les fausses alarmes donnant une action vraie alors que*

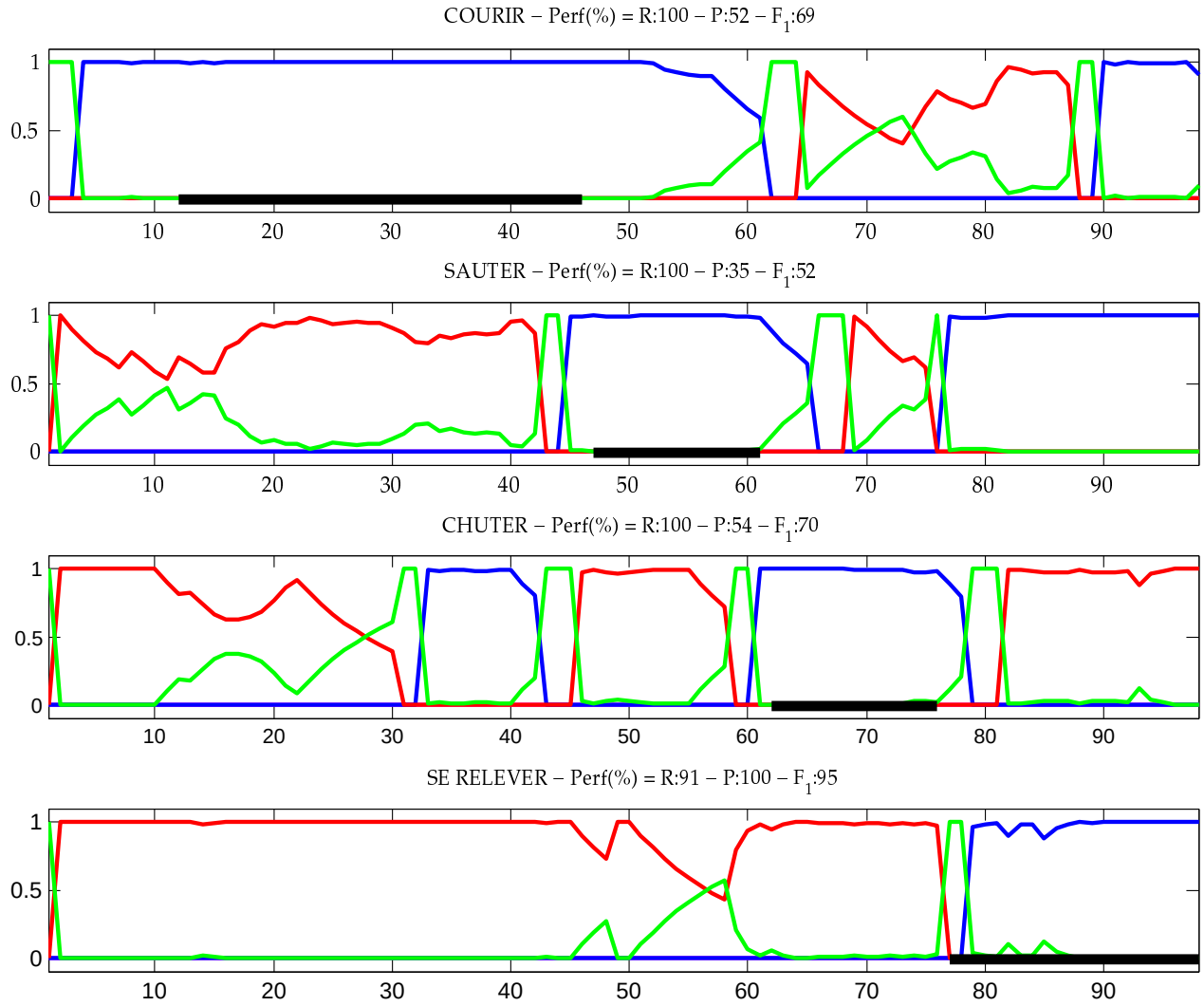



**FIG. 4.6 :** *Croyances sur les actions après filtrage.*

*l'état en cours est faux. Si le rappel n'est pas diminué, l'amélioration de la précision entraîne celle de la mesure $F_1$ comme le montre le tableau 4.1. L'indice de rappel peut être amélioré car le TCF et le CSS suppriment les fausses alarmes donnant une action fausse alors que l'état en cours est vrai. Notons aussi que le rappel peut augmenter si l'évaluation tient compte de la valeur de la croyance (ou de la pignistique) car le TCF et le CSS permettent de renforcer cette dernière. Enfin, d'un point de vue qualitatif, les croyances générées par le CSS sont beaucoup plus facilement interprétables.*

## 4.3 Mécanismes de préemption et de forçage pour synchroniser les états

Le point clé d'une méthode de reconnaissance de séquences est la façon de passer d'un état à un autre. Dans le cas idéal, les croyances sur les états fournies par les sources ou par le filtre TCF sont synchronisées, i.e. s'enchaînent parfaitement (Fig. 4.8a). Cependant, dans les applications réelles, les croyances sont généralement bruitées et les détections s'enchaînent moins naturellement (Fig. 4.8b) :

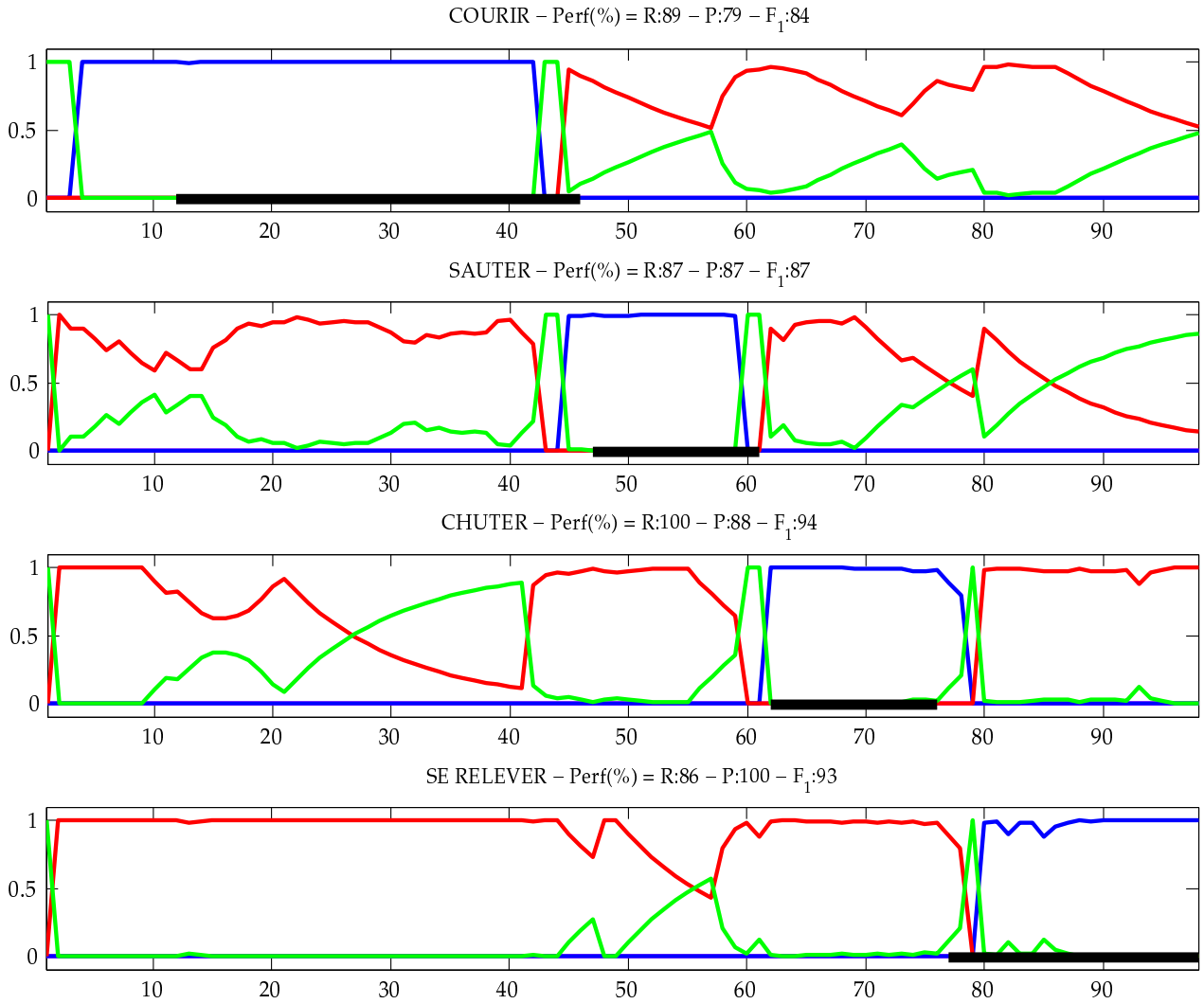



**FIG. 4.7 :** *Croyances sur les actions après filtrage et prise en compte des liens de causalité par l'ordonnanceur.*

| | course | | | saut | | | chute | | | se relever | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
| Sources | 100 | 49 | 66 | 100 | 33 | 50 | 100 | 47 | 64 | 77 | 81 | 79 |
| TCF | 100 | 52 | 69 | 100 | 35 | 52 | 100 | 54 | 70 | 91 | 100 | 95 |
| CSS | 89 | 79 | 84 | 87 | 87 | 87 | 100 | 88 | 94 | 86 | 100 | 93 |

**TAB. 4.1 :** *Indices de performance : rappel ($R$), précision ($P$) et mesure $F_1$ en % sur la détection des actions fournies par les sources (ligne 1), après application du filtre TCF (ligne 2) et après application de l'ordonnanceur CSS (ligne 3).*

elles se chevauchent ou laissent des espaces vides. On observe ces mêmes phénomènes sur les données réelles comme sur les figures 4.5 et 4.6 qui illustrent ce problème. Il est donc nécessaire de *synchroniser* ces états.

Nous proposons des mécanismes pour traiter le cas où deux états successifs dont l'état en cours sont vrais au même instant, et le cas où deux états successifs dont l'état en cours sont faux au même instant (Fig. 4.9) :

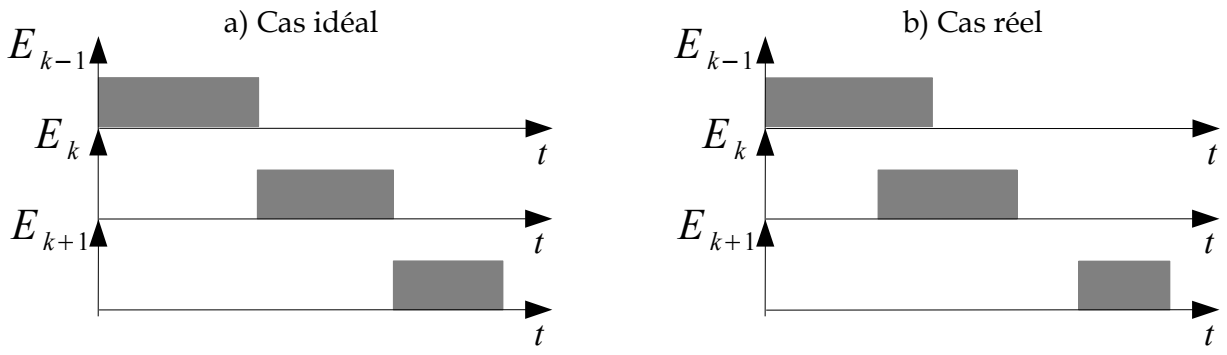



**FIG. 4.8 :** *Détection des états dans le cas idéal et dans le cas réel.*

- **Etats superposés :** à l'instant $t_P$, appelé instant de *préemption*, l'état $E_{k+1}$ devient *naturellement* vrai alors que $E_k$ est toujours naturellement vrai.
- **Etats disjoints :** à l'instant $t_F$, appelé instant de *forçage*, l'état $E_k$ devient naturellement faux alors que $E_{k+1}$ est faux. Donc deux états sont faux *naturellement*.

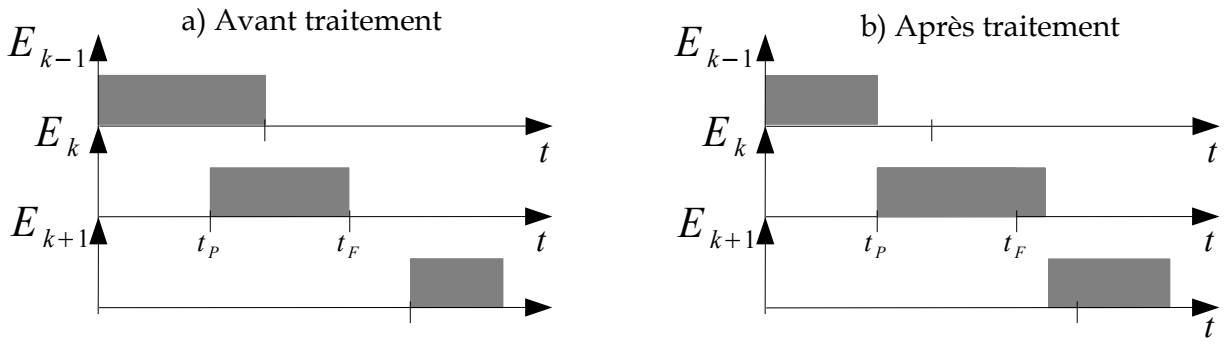



**FIG. 4.9 :** *Détection des états dans le cas idéal avant et après traitement par l'ordonnanceur.*

**Notations** Pour décrire les mécanismes, nous reprenons les notations du filtre TCF définies au chapitre 3 mais en ajoutant l'indice $k$ de l'état. Par exemple, la somme cumulée CUSUM de l'erreur sur l'état $E_k$ à l'instant $t$ est notée $CS^k(t)$ et son seuil de stop est noté $\mathcal{T}_s^k$. Le critère de qualité locale LQ défini au chapitre 3 (Eq. 3.21) pour un état $E_k$ dans une séquence $S_n$ est noté $LQ_{k,n}^{t_s:t}(h)$. On rappelle que $t_s$ est le début de l'état, $t$ est l'instant en cours et $h \in \{T_k^t, F_k^t\}$ est l'hypothèse (vrai ou faux) dont on cherche à évaluer la qualité.

### 4.3.1 Gestion des états superposés : *préemption*

La figure 4.10a illustre la superposition de deux états successifs $E_k$ et $E_{k+1}$. A l'instant de *préemption* $t_P$, deux stratégies sont possibles :

- $E_k$ reste vrai et on contraint $E_{k+1}$ à être faux,
- $E_{k+1}$ devient vrai et on contraint $E_k$ à être faux.

Nous avons choisi la deuxième stratégie car elle favorise l'évolution de la séquence. On obtient alors le résultat de la figure 4.10b.

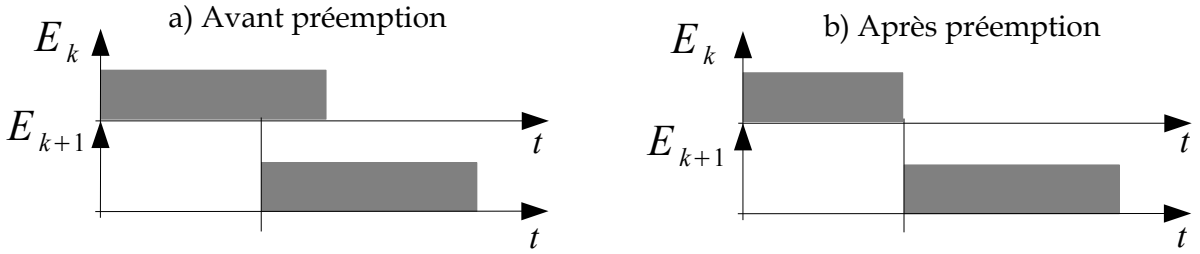



**FIG. 4.10 :** *Préemption : l'état $E_{k+1}$ prend naturellement le modèle « $\mathcal{T}$ : l'état est vrai » tandis que l'état $E_k$ est contraint à utiliser le modèle « $\mathcal{F}$ : l'état est faux ».*

Une préemption est détectée lorsque la somme cumulée du conflit $CS^{k+1}(t_P)$ de l'état $E_{k+1}$ dépasse son seuil de stop $\mathcal{T}_s^{k+1}$, i.e. $CS^{k+1}(t_P) > \mathcal{T}_s^{k+1}$ (état naturellement vrai).

Dans ce cas, deux modifications sont appliquées :

- **Modification 1 :** le filtre associé à l'état $E_{k+1}$ utilise naturellement le modèle « $\mathcal{T}$ : l'état est vrai » pour la prédiction,
- **Modification 2 :** le filtre associé à l'état $E_k$ est contraint à utiliser le modèle « $\mathcal{F}$ : l'état est faux » pour la prédiction.

On parle alors de *préemption* de l'état $E_{k+1}$ sur l'état courant $E_k$ à l'instant $t_P$.

**Fausse alarme** La qualité des croyances fournies par les sources peut être mauvaise lorsque l'on traite des données réelles. Dans ce cas, la détection de $E_{k+1}$ peut être une *fausse alarme*. Le contexte des filtres sur les états est sauvegardé avant la commutation des modèles et l'état $E_{k+1}$ est donc « temporairement » vrai. Nous verrons dans la section suivante comment nous validons ou invalidons une *préemption*.

**Etat d'initialisation** Nous définissons par ailleurs un état d'initialisation (état fictif situé en tête de séquence) qui est toujours vrai par défaut. Ainsi, l'ordonnanceur attend que le premier état, après l'état d'initialisation, soit vrai pour débuter. Une séquence débute donc toujours par une *préemption* du premier état sur l'état d'initialisation.

### 4.3.2 Gestion des états disjoints : *forçage*, et détection de fausses alarmes

Il existe plusieurs possibilités pour combler les trous entre les états naturellement faux. On considère la figure 4.11a où l'état en cours est $E_k$ (lorsque $t < t_F$).

Le mécanisme de *forçage* intervient à un instant $t_F$ lorsque :

- **Condition 1 :** la somme cumulée du conflit $CS^k(t_F)$ du filtre associé à l'état en cours $E_k$ indique que l'état devient naturellement faux, i.e. $CS^k(t_F) > \mathcal{T}_s^k$,
- **Condition 2 :** au même moment ($t_F$), le filtre associé à l'état suivant $E_{k+1}$ n'indique pas de changement, i.e. $CS^{k+1}(t_F) < \mathcal{T}_s^{k+1}$, et reste donc naturellement faux.

Ainsi, les deux états sont naturellement faux. Pour gérer ce problème, la stratégie employée consiste à contraindre le filtre de l'état $E_{k+1}$ à utiliser le modèle « $\mathcal{T}$ : l'état est vrai » dans le but de favoriser l'évolution de la séquence.

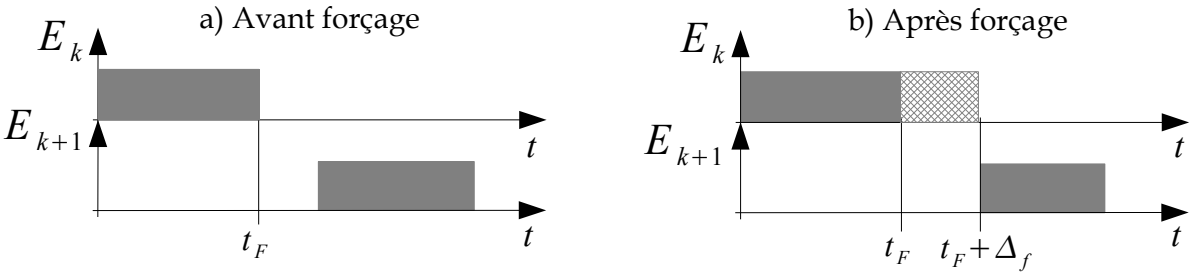



**FIG. 4.11 :** *Forçage : l'état $E_k$ utilise naturellement le modèle « $\mathcal{F}$ : l'état est faux » tandis que l'état $E_{k+1}$ est contraint à utiliser le modèle « $\mathcal{T}$ : l'état est vrai ».*

Afin de combler le « vide » entre les états faux, nous utilisons un nouveau paramètre $\Delta_F$ jouant le rôle d'une temporisation et qui permet, en fonction de la valeur imposée par un utilisateur, trois possibilités :

- **Fonctionnement général :** si $0 < \Delta_F < \infty$ alors le filtre sur l'état en cours $E_k$ attend une durée finie $\Delta_F$ pendant laquelle l'état suivant $E_{k+1}$ peut devenir vrai. Au delà de cette durée, le filtre sur l'état suivant $E_{k+1}$ est contraint à utiliser le modèle « $\mathcal{T}$ : l'état est vrai » à moins que le filtre sur l'état $E_k$ qui demandait à changer de modèle ne le souhaite plus, i.e. $CS^k(t) < \mathcal{T}_s^k$. Ce cas est illustré sur la figure 4.11b.
- **Forçage immédiat :** si $\Delta_F = 0$ alors le filtre sur l'état $E_{k+1}$ est contraint à utiliser immédiatement le modèle « $\mathcal{T}$ : l'état est vrai » tandis que le filtre sur $E_k$ utilise naturellement le modèle « $\mathcal{F}$ : l'état est faux ».
- **Attente d'une préemption :** si $\Delta_F = \infty$ alors l'état en cours $E_k$ est contraint à conserver le modèle vrai tant que l'état suivant $E_{k+1}$ n'est pas vrai, i.e. tant que $CS^{k+1}(t) < \mathcal{T}_s^{k+1}$.

Suite à un forçage de $E_k$ sur $E_{k+1}$, l'ordonnanceur CSS propose deux possibilités :

1) **$\mathbf{E_k}$ avait exécuté une *préemption* sur $\mathbf{E_{k-1}}$ :** si l'état qui impose le *forçage*, i.e. $E_k$, avait préalablement exécuté une *préemption* sur $E_{k-1}$ alors un mécanisme de validation ou d'invalidation de la *préemption* (§ 4.3.1) intervient. Ce mécanisme est illustré sur la figure 4.12. La *préemption* est validée si la qualité de l'état effectuant le *forçage*, i.e. $E_k$, est suffisante. Cette qualité est quantifiée par le critère de qualité locale $LQ_{k+1,n}^{t_P:t_F+\Delta_F}(T^f)$ décrit dans le paragraphe 3.4. Deux cas se présentent :

   a) **Invalidation de la préemption $\Rightarrow$ $\mathbf{E_{k-1}}$ redevient l'état en cours :** si la qualité n'est pas suffisante, i.e. elle est inférieure à un seuil $\delta_{\text{FA}}$ ($LQ_{k+1,n}^{t_P:t_F+\Delta_F}(T^f) < \delta_{\text{FA}}$), alors la détection de $E_k$ sur l'intervalle $[t_P, t_F + \Delta_F]$ est une *fausse alarme* (Fig. 4.13a). Par conséquent, la préemption de $E_k$ n'est pas validée et le forçage sur $E_{k+1}$ est annulé. Le contexte de $E_{k-1}$, préalablement sauvé lors de la préemption (§ 4.3.1), est restitué : l'état en cours est de nouveau $E_{k-1}$ et le filtre sur $E_{k-1}$ est contraint d'utiliser le modèle « $\mathcal{T}$ : l'état est vrai » sur l'*intervalle de fausse alarme* $[t_P, t_F + \Delta_F]$. Le filtre sur $E_k$ est par ailleurs contraint à utiliser le modèle « $\mathcal{F}$ : l'état est faux » sur cet intervalle. Le résultat est illustré sur la figure 4.13b et sur la figure 4.12 (en haut à droite).

   b) **Validation de la préemption $\Rightarrow$ $\mathbf{E_{k+1}}$ devient l'état en cours :** si la qualité est suffisante

($LQ_{k+1,n}^{t_P:t_F+\Delta_F}(T^f) > \delta_{\mathrm{FA}}$) alors le filtre sur l'état $E_{k+1}$ est contraint à utiliser le modèle « $\mathcal{T}$ : l'état est vrai » tandis que l'état $E_k$ utilise naturellement le modèle « $\mathcal{F}$ : l'état est faux ». La préemption est donc *validée*. Ce cas est illustré sur la figure 4.12 (en bas à droite).

2) **$\mathbf{E_{k-1}}$ avait exécuté un *forçage* sur $\mathbf{E_k}$ :** si le filtre sur l'état qui exécute un *forçage* ($E_k$) avait préalablement subit un *forçage* (par $E_{k-1}$) alors le filtre sur l'état $E_{k+1}$ est contraint à utiliser le modèle « $\mathcal{T}$ : l'état est vrai » tandis que le filtre sur l'état $E_k$ utilise naturellement le modèle « $\mathcal{F}$ : l'état est faux ».

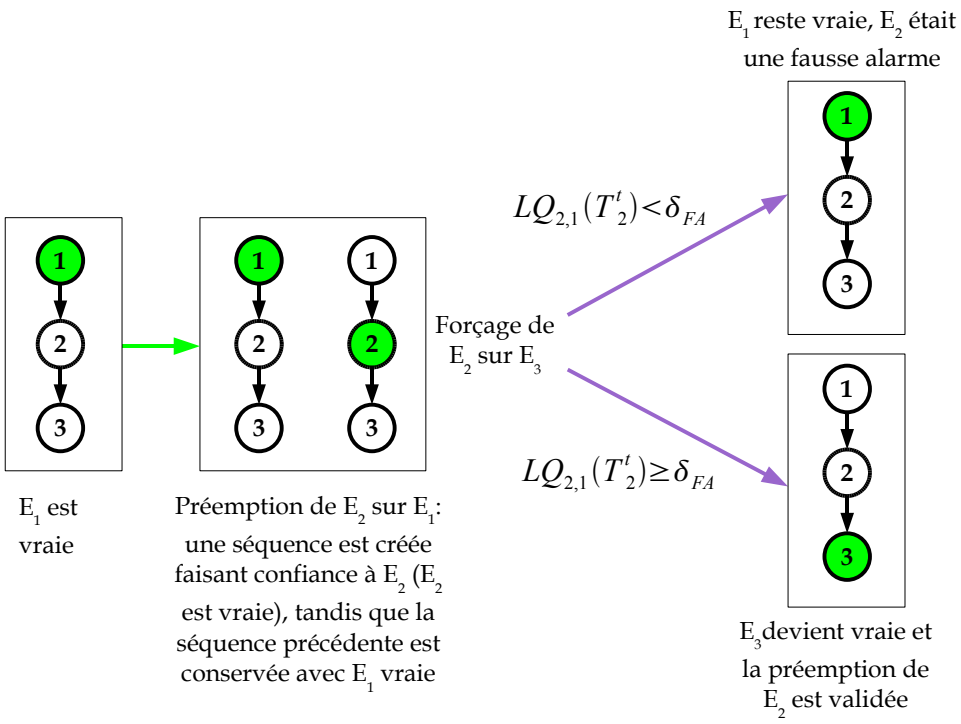



**FIG. 4.12 :** *Détection des fausses alarmes : le critère de qualité locale $LQ_{2,1}(T^f)$ représente la qualité de la proposition* « état $E_2$ est vrai » *conditionnellement au modèle vrai dans la séquence $S_1$.*

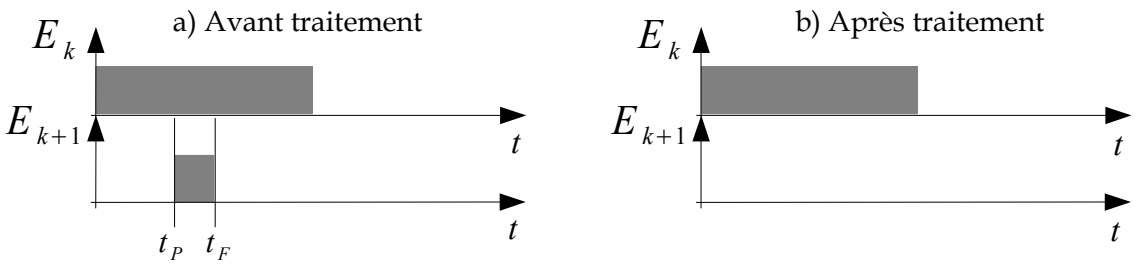



**FIG. 4.13 :** *Fausse alarme : une préemption suivie d'un forçage indique une fausse alarme potentielle.*

Les mécanismes de l'ordonnanceur permettent de suivre une évolution sur les états composant la séquence en garantissant une certaine cohérence entre le modèle de la séquence et les données.

## 4.4 Inférence et classe de rejet

Il s'agit à présent de reconnaître une séquence parmi plusieurs possibles, chacune d'elles étant décrite par un ordonnanceur CSS. Pour cela, il est nécessaire de disposer d'un critère appelé *critère d'inférence* pour quantifier la qualité d'une séquence reconnue. Un tel critère permet notamment de choisir la meilleure séquence, « meilleure » au sens du critère choisi, parmi un ensemble de séquences possibles permettant ainsi de résoudre des problèmes de classification. Dans ce paragraphe, nous présentons un critère d'inférence permettant de mesurer, en ligne, la qualité d'une séquence dans l'ordonnanceur CSS.

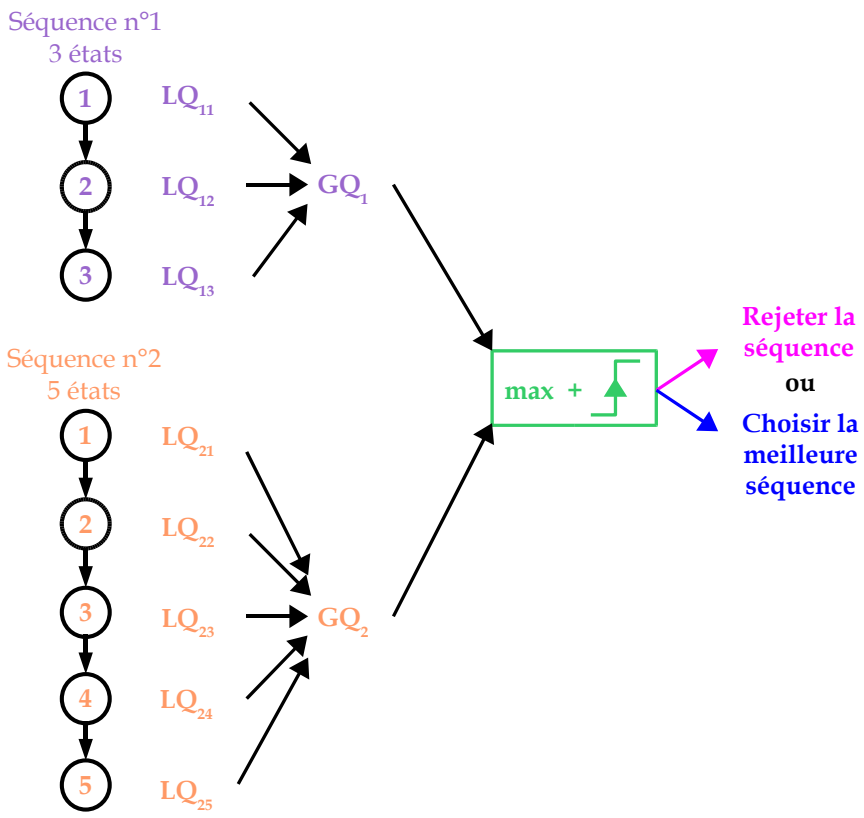



**FIG. 4.14 :** *Inférence réalisée en ligne par agrégation des qualités locales.*

Le critère que nous proposons est nommé *critère global de reconnaissance* (GQ). Son calcul est basé sur l'utilisation du critère local défini pour un état dans le paragraphe 3.4. Nous rappelons que ce critère local est noté $LQ_{i,n}^{t_s:t}(T^t)$ et représente la qualité du filtrage de l'état $E_i$ lorsque le modèle « $\mathcal{T}$ : l'état est vrai » est utilisé, i.e. $E_i$ est l'état en cours.

Le principe du critère est illustré sur la figure 4.14. Pour une séquence $S_n$ composée de $K_n$ états, nous proposons d'agréger les critères locaux $LQ_{i,n}^{t_s:t}(T^t)$ pour tous les états $E_i$ dans $S_n$. Le résultat est noté $GQ_n^t$ qui représente la qualité de reconnaissance de la séquence $S_n$ sur l'intervalle de temps $[t_s, t]$ où $t_s$ est l'instant de départ (pouvant être soit un instant de préemption $t_P$ de l'état courant sur l'état précédent ou un instant de forçage $t_F$ de l'état précédent sur l'état courant) et où $t$ est l'instant courant.

La fonction d'agrégation utilisée ici est la moyenne des critères locaux sur l'ensemble des états

de la séquence :

$$GQ_n^t = \frac{1}{K_n} \sum_{i \in \{1..K_n\}} LQ_{i,n}^{t_s:t}(T^t) \tag{4.3}$$

Ainsi, disposant de $N$ séquences ($n \in \{1 \dots N\}$), la séquence $S_*^t$ qui correspond le mieux aux mesures à l'instant courant $t$ est celle qui maximise le critère $GQ_n^t$ sur l'ensemble des séquences possibles (Fig. 4.14) :

$$S_*^t = S_n^t \quad \text{si} \quad GQ_n^t = \max_{n \in \{1 \dots N\}} (GQ_n^t > \theta) \tag{4.4}$$

où $\theta$ est un degré de sévérité sur la qualité. Ce paramètre permet de créer une *classe de rejet* si la meilleure séquence est de mauvaise qualité. La classe de rejet est importante en classification. Elle permet de diminuer les erreurs mais aussi d'attirer l'attention sur des cas « inconnus » et donc peut permettre une éventuelle adaptation (par exemple la création ou la suppression d'états dans une séquence). Créer une séquence de rejet avec l'ordonnanceur est possible du fait que $GQ$ est borné sur $[0, 1]$ (les critères locaux étant bornés sur $[0, 1]$, la moyenne est aussi bornée sur cet intervalle). Notons que pour éviter de multiplier le nombre de paramètres, la valeur de $\theta$ peut être la même que celle de $\delta_{\text{FA}}$, i.e. le seuil sur la valeur minimale de qualité de l'état.

Comme souligné dans le paragraphe 3.4, le critère $LQ$ peut être basé sur la plausibilité $pl$, la crédibilité $bel$ ou la pignistique $\text{BetP}$ avec la relation d'ordre donnée par l'équation 3.22. Cette relation peut être réécrite avec le critère $GQ$ à savoir $GQ_{\text{BetP}}^t, GQ_{bel}^t, GQ_{pl}^t \in [0, 1]$ et :

$$GQ_{bel}^t \leq GQ_{\text{BetP}}^t \leq GQ_{pl}^t \tag{4.5}$$

La figure 4.15 illustre l'évolution du critère $GQ$ pour les variantes : $GQ_{\text{BetP}}^t$, $GQ_{bel}^t$ et $GQ_{pl}^t$, lors de la reconnaissance d'un saut en hauteur. Au cours du temps, la valeur du critère augmente jusqu'à environ $92\%$ signifiant que la *qualité* de la séquence reconnue est de $92/100$. Ce critère, qui est calculé en ligne, est basé sur le conflit rencontré au cours de la séquence lors des prédictions sur les différentes états de la séquence. On remarque que les trois variantes tendent à se rapprocher et ce qui est normal puisque toutes les trois tendent vers l'unité au fur et à mesure que la séquence évolue.

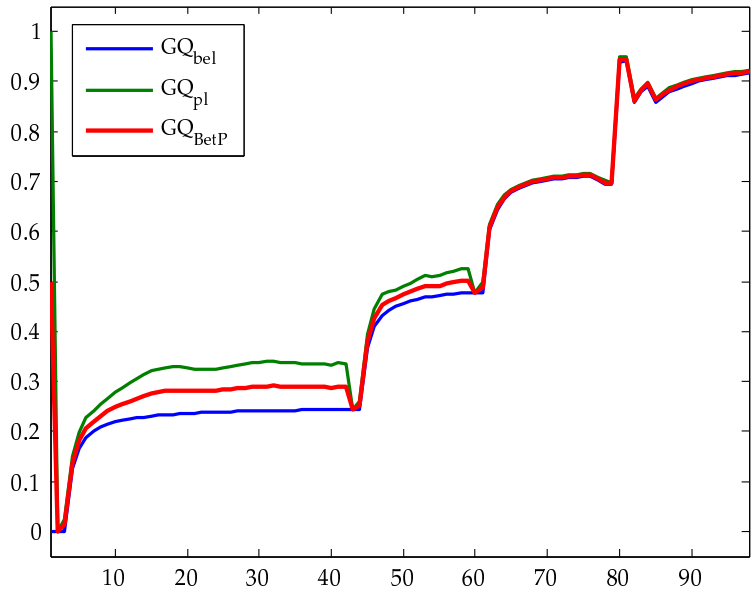



**FIG. 4.15 :** *Evolution du critère d'inférence au cours de la reconnaissance de la séquence d'états.*

**Exemple 22.** *Nous considérons un problème de classification entre* $4$ *classes appelées* activités *:* $S_1$ *:* saut en hauteur, $S_2$ *:* saut en longueur, $S_3$ *:* saut à la perche *et* $S_4$ *:* triple saut. *Chaque activité est décrite par une séquence d'actions et chacune des activités est reconnue par un ordonnanceur. Nous supposons disposer des détecteurs d'actions. L'objectif est de classer* $26$ *vidéos de* sauts à la perche *: chaque vidéo est analysée par les détecteurs d'actions de chaque activité et par les* $4$ *ordonnanceurs. Chaque ordonnanceur génère une valeur pour le critère* $GQ_i, i \in \{1 \dots 4\}$ *quantifiant la qualité de la reconnaissance de l'activité* $S_i$ *à chaque instant pour la vidéo considérée.*

*La figure 4.18 récapitule les valeurs des critères globaux* $GQ$ *pour chaque ordonnanceur, les valeurs reportées correspondant aux valeurs calculés à la fin de chaque vidéo. Par exemple, pour la deuxième vidéo, on obtient* $GQ_1 \approx 36\%$ *pour le* saut en hauteur, $GQ_2 \approx 0\%$ *pour le* saut en longueur, $GQ_3 \approx 70\%$ *pour le* saut à la perche *et* $GQ_4 \approx 32\%$ *pour le* triple saut. *La valeur maximum est obtenue pour la séquence* $S_3$ *correspondant au* saut à la perche *et ce résultat est en accord avec la vérité terrain. Les erreurs de classification sont soulignées avec un point rouge : c'est le cas pour les vidéos* $7$ *et* $8$ *donnant le* saut en hauteur *gagnant. La vidéo 8 est illustrée sur la figure 4.16 qui montre que l'athléte est d'abord filmé par l'arrière puis de côté.*

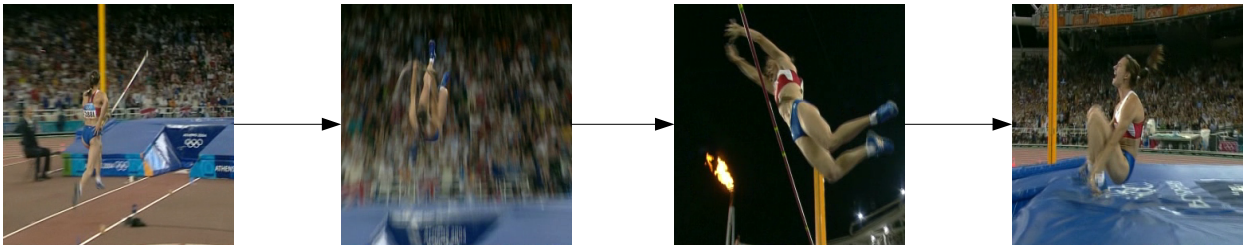



**FIG. 4.16 :** *Le saut à la perche de la vidéo 8 présente un angle de vue qui rend difficile la reconnaissance du saut.*

*Les rejets sont affichés avec des points bleus. Lorsque le maximum des critères est en dessous du seuil* $50\%$*, la décision est rejetée : c'est le cas pour des vidéos correctement classées comme les vidéos* $13$ *et* $19$*, mais c'est aussi le cas pour les erreurs de classification comme dans le cas de la vidéo* $24$ *(Fig. 4.17). Cette vidéo est difficile à reconnaître à cause de l'angle de vue qui est très différent de celui des autres vidéos.*

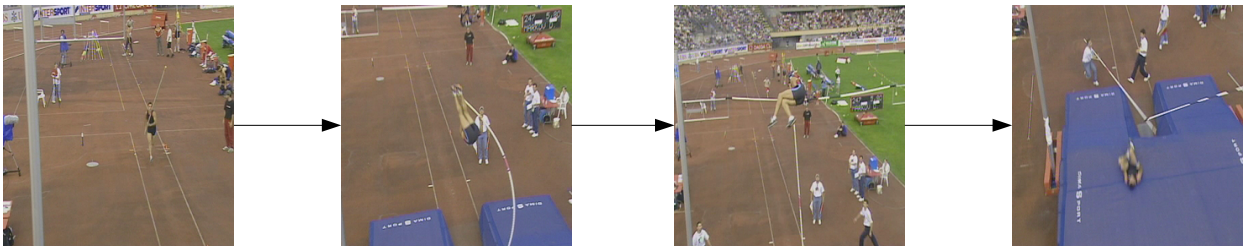



**FIG. 4.17 :** *Le saut à la perche de la vidéo 24 présente un angle de vue qui rend difficile la reconnaissance du saut.*

*A titre comparatif, nous donnons à la figure 4.19 les valeurs des log-vraisemblances de ces mêmes séquences dans le cadre de la reconnaissance par modèle de Markov caché (HMM) [Rab89]. Un*

*HMM est construit pour chaque type d'activités avec le même nombre d'états que les ordonnanceurs. A la différence de l'ordonnanceur, un HMM autorise des transitions entre des états qui ne sont pas forcément consécutifs. Pour la classification des vidéos, nous appliquons l'algorithme* forward *qui calcule la log-vraisemblance de la séquence des observations (fournies par les détecteurs d'actions) et ce résultat est reporté sur la figure 4.19 (le type de vidéo est choisi en* maximisant *la log-vraisemblance sur les différents HMM). Les résultats montrent que les log-vraisemblances sont relativement proches et donc que la décision est peu fiable. De plus, il est difficile de créer une classe de rejet étant donné que les log-vraisemblances peuvent tendre vers la valeur* $-\infty$.

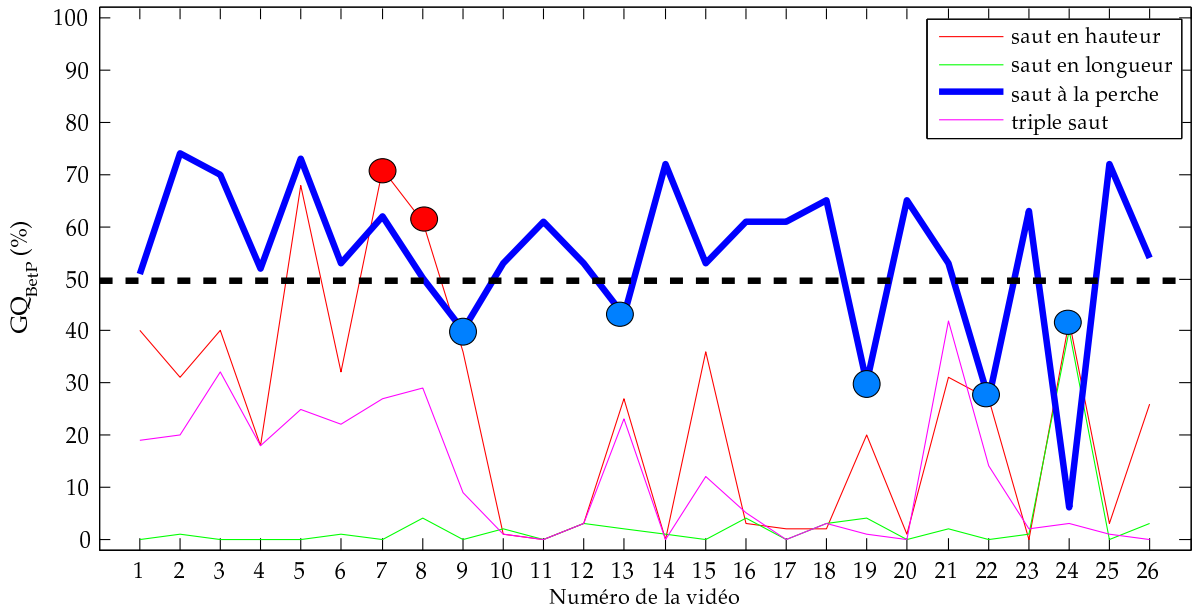



**FIG. 4.18 :** *Valeur des critères de reconnaissance de séquences GQ pour* 26 *vidéos de* saut à la perche *en utilisant les* 4 *ordonnanceurs CSS.*

*La figure 4.20 illustre l'évolution des critères des* 4 *ordonnanceurs pour un* saut à la perche *(vidéo numéro 2 de la figure 4.19). Le critère GQ étant calculé « en ligne », il peut être utilisé pour classer des systèmes dynamiques au cours du temps et connaître à quel niveau de la séquence le système se trouve ainsi que le meilleur état en cours.*

## 4.5 Conclusion

La machine à états proposée dans ce chapitre permet de reconnaître des séquences où chaque état est décrit par une fonction de croyance binaire. Au lieu de gérer une distribution sur l'ensemble des états comme pour le réseau de Pétri crédibiliste, l'ordonnanceur CSS proposé utilise autant de croyances binaires que d'états, les fonctions de croyances sont ainsi plus simples. Les mécanismes de transition sont basés sur des changements de modèles avec validation ce qui rend l'ordonnanceur CSS plus robuste que le réseau de Pétri disponible dans la littérature mais en contrepartie, cela oblige à régler les seuils de commutation. Le critère de validation que nous avons proposé pour les changements de modèles sur les états exploite le conflit entre mesures et modèles. Ce critère est par ailleurs

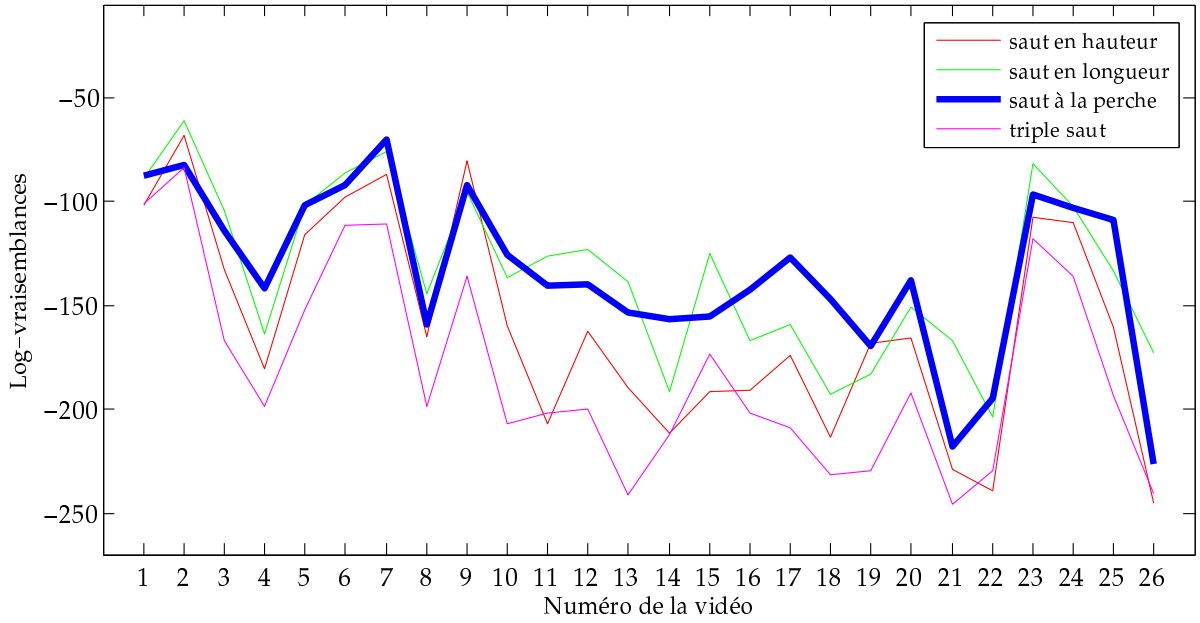



**FIG. 4.19 :** *Valeur des log-vraisemblances en sortie des* 4 *HMM (un pour chaque saut) pour la reconnaissance de séquences de* 26 sauts à la perche.

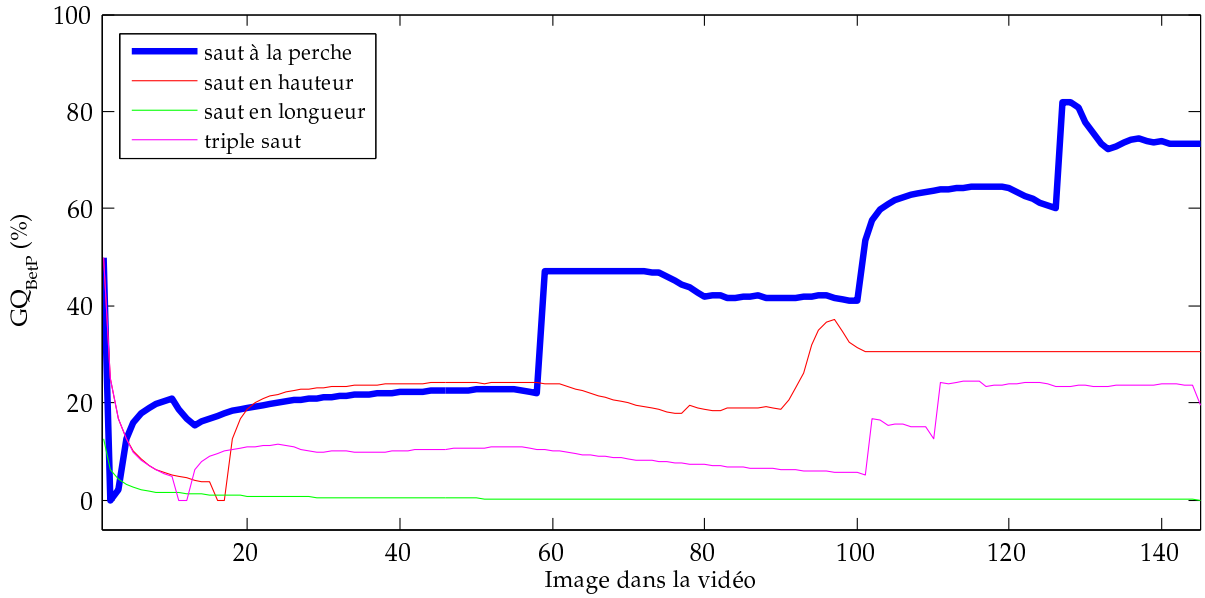



**FIG. 4.20 :** *Evolution du critère GQ dans l'ordonnanceur pour la reconnaissance en ligne d'un saut à la perche (vidéo numéro 2 de la figure 4.19).*

exploité pour reconnaître des séquences en ligne et il permet notamment de créer une classe de rejet ce qui le rend intéressant pour l'adaptation de séquences.

Les limitations concernent d'une part la décision stricte sur les transitions. Même si elle est parfois différée, une décision plus « floue » serait plus aisée à régler par un utilisateur. Cependant, cela com-

plexifie le mécanisme de reconnaissance. D'autre part, nous supposons que les modèles de séquence sont connus or, il serait intéressant de disposer d'une méthode de détermination automatique.

Dans le chapitre suivant, nous proposons les premières pistes pour généraliser les Modèles de Markov Cachés probabilistes aux fonctions de croyances. Ce type d'outil permet d'apprendre automatiquement la structure des séquences et requiert non plus une connaissance experte mais des données a priori sous la forme d'une base d'apprentissage.

# Chapitre 5

# Reconnaissance non-déterministe de séquences d'états par modèles de Markov cachés crédibilistes

## 5.1 Introduction

Dans certaines applications, on ne dispose pas d'information a priori sur les modèles des séquences mais seulement de séquences d'observations numériques, continues ou discrètes, bruitées. Dans ce cas, il est nécessaire de disposer d'algorithmes pour :

- estimer la topologie des séquences (nombre d'états et liens entre états) à partir des observations,
- lier les observations numériques aux états symboliques.

Des outils performants basés sur le Traitement du Signal et les Statistiques ont été proposés pour régler ces problèmes dans le cadre de la théorie des probabilités. En particulier, les *modèles de Markov cachés* (HMM : *Hidden Markov Models*) [Rab89] représentent l'une des méthodes les plus populaires ces 30 dernières années avec de nombreuses applications comme nous l'avons vu dans l'introduction du manuscrit.

Par rapport à l'ordonnanceur CSS présenté dans le chapitre précédent, on dispose à présent non plus de fonctions de croyance binaires sur chacun des états mais de fonctions de croyance sur l'ensemble des états. De plus, dans le cadre des HMM, un état peut être non seulement relié à un état amont et un état aval mais aussi à tous les autres états et le passage d'un état à un autre n'est plus déterministe mais stochastique.

Après avoir présenté les HMM probabilistes, nous proposons une extension des algorithmes des HMM afin de pouvoir les utiliser sur des fonctions de croyance. L'objectif d'une telle extension est de profiter de la généricité des fonctions de croyance et de l'efficacité des algorithmes des HMM. Nous montrerons des exemples d'application sur l'analyse de mouvements humains dans les vidéos de sports.

## 5.2 HMM probabiliste

La description des HMM probabilistes proposée dans ce chapitre est basée sur le tutoriel de Rabiner [Rab89]. Nous commençons par présenter les concepts de base et les notations. Puis, nous

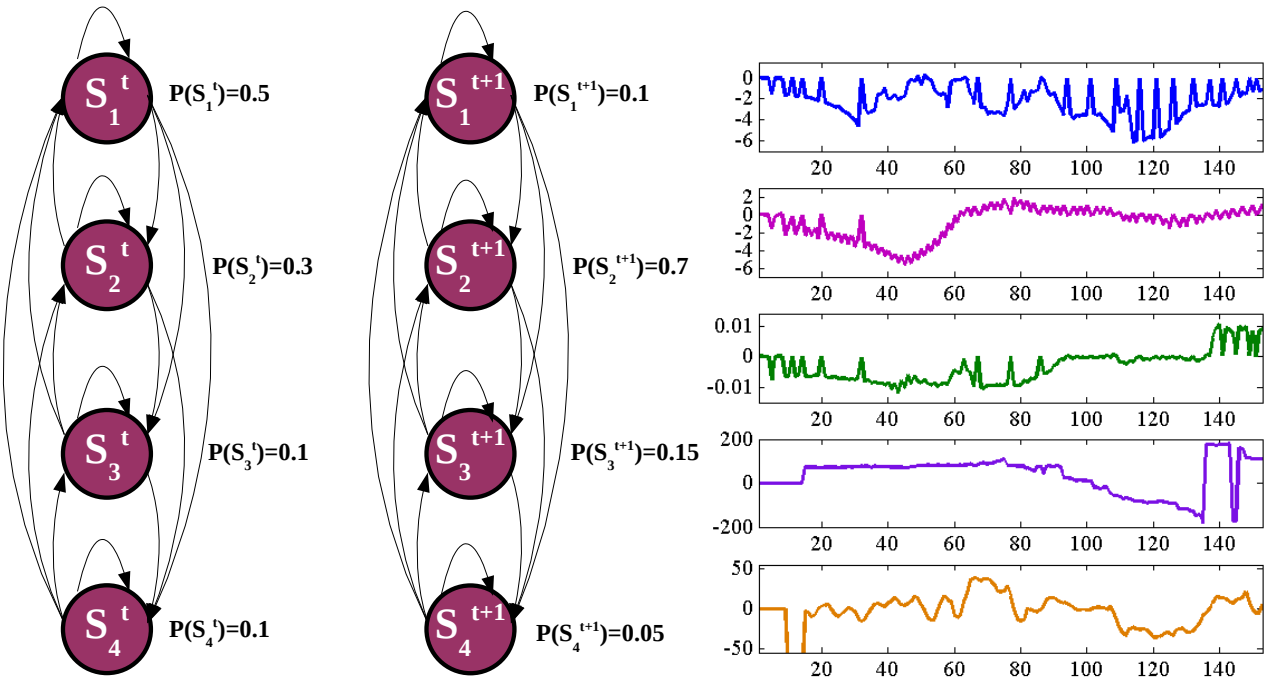
détaillons les mécanismes probabilistes permettant de propager les distributions de probabilités sur les états tout au long d'une séquence et nous présentons quelques algorithmes d'analyse de séquences basés sur ces mécanismes.

### 5.2.1 Modèle de séquence et observations

Dans le contexte des HMM, une séquence est définie par un ensemble de $K$ états reliés par des arcs orientés comme illustré sur la figure 5.1a. Les états sont notés :

$$\Omega_t = \{s_1^t, s_2^t \dots s_i^t \dots s_j^t \dots s_K^t\} \tag{5.1}$$

où l'*indice* $j$ représente le numéro de l'état et l'*exposant* $t$ l'instant courant.




a) Un HMM composé de 4 états à deux instants consécutifs.

b) Des observations continues fournies par des sources.

**FIG. 5.1 :** *Un HMM composé de $K$ états à deux instants consécutifs et des observations* numériques *continues (5 sources d'information) évoluant au cours du temps qu'il s'agit de lier aux états. Les unités de ces observations n'ont pas d'importance ici.*

Au fil du temps, le processus observé passe d'un état à un autre avec une probabilité donnée modifiant ainsi la distribution de probabilités sur les états (Fig. 5.1a). Un des problèmes de la reconnaissance de séquences dans le cadre des HMM est de déterminer une séquence d'états optimale à partir d'observations mesurées sur le système (Fig. 5.1b).

Les états sont dits *cachés* car seules les observations sont effectivement mesurées. Aux instants $t \in \{1 \dots T\}$, on dispose d'observations numériques, supposées continues, pouvant être bruitées et reflétant l'état du système dynamique étudié. Une observation à l'instant $t$, notée :

$$\mathbf{O}_t = [O_1, O_2 \dots O_F], \mathbf{O}_t \in \Re^F \tag{5.2}$$

est composée de $F$ caractéristiques numériques extraites sur le système dynamique à étudier. Le lien entre les observations, généralement bruitées, et les $K$ états dans $\Omega_t$ d'un HMM $\lambda$ est défini par une

transformation dite « numérique-symbolique ». Une observation $\mathbf{O}_t$ génère un degré de confiance sur chaque état prenant généralement la forme, dans le cadre des HMM probabilistes, de vraisemblances $P(\mathbf{O}_t|s_i^t)$ et que l'on note généralement $b_i(\mathbf{O}_t)$ [Rab89]. L'ensemble des modèles d'observations permettant de transformer une valeur numérique observée en vraisemblance $b_i(\mathbf{O}_t)$ sur les états, est classiquement noté $B$.

Un HMM autorise des liens de causalité entre des états qui ne sont pas forcément consécutifs à la différence de l'ordonnanceur présenté dans le chapitre précédent. A chaque instant, la transition d'un état $s_i^{t-1}$ à un état $s_j^t$ se fait à l'aide d'une probabilité :

$$a_{ij} = P^{\Omega_t}(s_j^t|s_i^{t-1}) \tag{5.3}$$

La matrice de transition $A = [a_{ij}], i \in \{1 \dots N\}, j \in \{1 \dots N\}$, de taille $N \times N$ est constituée des probabilités de transition et cette matrice est *stochastique*, i.e. la somme des probabilités sur les colonnes est unitaire, i.e. $\sum_j a_{ij} = 1$ et $a_{ij} > 0$. Lorsque l'état courant est connu, l'état suivant n'est pas déterminé précisément mais par une distribution de probabilités, c'est pourquoi un HMM est une machine à états finis *probabiliste* et *non déterministe* [Mur95, Mur02]. Notons que l'on considère que le système étudié est Markovien[1] ce qui signifie qu'une distribution de probabilités à un instant $t$ peut être calculée uniquement à partir des informations à $t-1$ et à $t$.

Un HMM $\lambda$ est donc caractérisé par trois éléments :
- la matrice de transition $A = [a_{ij}], i \in \{1 \dots N\}, j \in \{1 \dots N\}$,
- les modèles d'observation $B$ générant les vraisemblances $b_i(\mathbf{O}_t), i \in \{1 \dots N\}, t \in \{1 \dots T\}$,
- la distribution initiale $\pi_i, i \in \{1 \dots N\}$ sur les états à $t = 1$.

Un HMM est généralement noté $\lambda = \{A, B, \Pi\}$ où $A$, $B$ et $\Pi$ représentent respectivement l'ensemble des distributions conditionnelles $a_{ij}$, des vraisemblances conditionnelles $b_i(\mathbf{O}_t)$ et des probabilités a priori $\pi_i$.

A partir d'une séquence d'observations $\mathbf{O}_{1:T}$ de longueur $T$, le cadre formel des HMM offre alors des algorithmes permettant de :

1. **Calculer la probabilité sur les différents états de la séquence** à partir de la distribution sur les états à l'instant précédent, des probabilités de transition entre les états et des vraisemblances sur les états,
2. **Trouver la meilleure séquence d'états** à partir de la vraisemblance des états,
3. **Déterminer le meilleur modèle de HMM** parmi un ensemble de HMM possibles à partir des vraisemblances sur les états (problème de *classification de séquences*),
4. **Estimer les paramètres d'un HMM** à partir d'un ensemble d'apprentissage composé d'observations.

Pour chacun de ces quatre problèmes, une solution sera présentée dans les paragraphes suivants. Ces solutions reposent, comme nous le verrons, sur les méthodes de *propagation forward* (ce qui signifie *vers l'avant*, de l'instant $t = 1$ à $T$) et *backward* (ce qui signifie *vers l'arrière*, de l'instant $T$ à $t = 1$) [Rab89, Mur02] de distributions de probabilités[2] que nous présentons dans le paragraphe suivant.

[1] Cette hypothèse est généralement admise dans la plupart des méthodes d'analyse de séquences [Mur02].
[2] Les propagations sont analogues à celles définies dans la section 2.2.1.3 pour les réseaux évidentiels.

### 5.2.2 Mécanismes de base : propagations *forward* et *backward*

Les propagations *forward* et *backward* permettent de simplifier le calcul de distributions de probabilités à partir de distributions conditionnelles grâce à des mécanismes de combinaison – projection (voir références dans [Mur02], voir aussi [KFL01] qui concerne le formalisme très générique des graphes factorisés). Dans le cadre des HMM, ces propagations permettent d'assurer une cohérence temporelle.

#### 5.2.2.1 Propagation *forward*

La propagation *forward* (Fig. 5.2a) est un processus markovien qui permet d'estimer, en ligne, la vraisemblance des états à un instant $t$ ne disposant que des éléments suivants :

- les vraisemblances $\alpha_{t-1}(s_i^{t-1})$ des états $s_i^{t-1} \in \Omega_{t-1}$ à $t-1$,
- les transitions $a_{ij}$ entre deux états $s_i^{t-1} \in \Omega_{t-1}$ et $s_j^t \in \Omega_t$,
- les vraisemblances sur les états $b_i(\mathbf{O}_t)$ générées par les observations courantes $\mathbf{O}_t$.

**FIG. 5.2 :** *Modèle graphique représentant les propagations* forward *et* backward *entre deux instants et où $\Omega_t$ est l'ensemble des états.*

La propagation *forward* permet de calculer la variable *forward* $\alpha_t(s_j^t)$ qui représente la vraisemblance d'un état $s_j^t \in \Omega_t$ et des observations partielles $\mathbf{O}_{1:t}$ conditionnellement au modèle de HMM $\lambda$ utilisé :

$$\alpha_t(s_j^t) = P(O_1, O_2 \ldots O_t, s_j^t|\lambda), \quad \forall s_j^t \in \Omega_t \tag{5.4}$$

L'estimation de cette vraisemblance est réalisée par la propagation *forward* qui respecte le schéma « prédiction – mise à jour » (Fig. 5.2a) :

**Etape 1 : calcul de la prédiction $\hat{\alpha}_t(s_j^t)$ :** La prédiction de la probabilité sur $s_j^t$ est obtenue par application de l'équation de conditionnement (chapitre 2, Eq 2.21) en utilisant les vraisemblances $\alpha_{t-1}(s_i^{t-1})$ sur l'ensemble des états à l'instant précédent ainsi que les transitions des états $s_i^{t-1}$ vers les états $s_j^t$ :

$$\hat{\alpha}_t(s_j^t) \quad = \sum_{s_i^{t-1} \in \Omega_{t-1}} \alpha_{t-1}(s_i^{t-1}) \cdot a_{ij} \tag{5.5}$$

A $t = 1$, on ne dispose que des probabilités a priori $\pi_{s_j^1}$ des états $s_j^1$ ainsi que des vraisemblances $b_i(\mathbf{O}_1)$. Il suffit alors de les combiner conjonctivement : $\forall s_i^1 \in \Omega_1, \alpha_1(s_i^1) = \pi_{s_i^1} \cdot b_i(\mathbf{O}_1)$.

**Etape 2 : mise à jour de $\alpha_t(s_j^t)$ :** La prédiction est combinée conjonctivement avec les vraisemblances sur les états. Dans le cadre probabiliste, la combinaison conjonctive est réalisée par

l'opérateur produit :

$$\alpha_t(s_j^t) = \hat{\alpha}_t(s_j^t) \cdot b_j(\mathbf{O}_t) \tag{5.6}$$

La procédure est récursive car la probabilité $\alpha_t(s_i^t)$ représente la mise à jour de la probabilité sur les états à $t$ qui sera utilisée à $t+1$ :

$$\hat{\alpha}_t(s_j^{t+1}) = \sum_{s_i^t \in \Omega_t} \alpha_t(s_i^t) \cdot a_{ij} \tag{5.7}$$

Par conséquent, la variable *forward* $\alpha_t(s_j^t) = P(O_1, O_2 \ldots O_t, s_j^t|\lambda)$ peut être calculée pour $t = 1$ à $t$ avec l'équation récursive suivante :

$$\alpha_t(s_j^t) = \left( \sum_{\forall s_i^{t-1} \in \Omega_{t-1}} \alpha_{t-1}(s_i^{t-1}) \cdot a_{ij} \right) \cdot b_j(\mathbf{O}_t) \tag{5.8}$$

Remarquons que pour obtenir une distribution, il est nécessaire de normaliser les vraisemblances $\alpha_t(s_j^t)$ par $\sum_j \alpha_t(s_j^t)$.

Le calcul de cette équation est illustré sur la figure 5.3 qui illustre le fait que le calcul de la probabilité $\alpha_t(s_j^t)$ sur un état $s_j^t$ prend en compte l'ensemble des *contributions* des états à l'instant précédent pondérées par les transitions. Une application de la propagation *forward* pour le filtrage d'une distribution de probabilités temporelle sur un cas réel (reconnaissance de mouvements humains dans une vidéo) est représentée sur la figure 5.4b (voir l'exemple 23 pour plus de détails).

#### 5.2.2.2 Propagation *backward*

La propagation *backward* (Fig. 5.2b) est un processus markovien qui permet d'estimer, hors ligne, la vraisemblance des états à un instant $t$ ne disposant que des éléments suivants :

- les vraisemblances $\beta_{t+1}(s_j^{t+1})$ des états $s_i^{t+1} \in \Omega_{t+1}$ à $t+1$,
- les transitions $a_{ij}$ entre deux états $s_i^{t+1} \in \Omega_{t+1}$ et $s_j^t \in \Omega_t$,
- les vraisemblances sur les états $b_i(\mathbf{O}_{t+1})$ générées par les observations futures $\mathbf{O}_{t+1}$.

La propagation *backward* permet de calculer la variable *backward* $\beta_t(s_j^t)$ qui représente la vraisemblance des observations partielles $\mathbf{O}_{t+1:T}$ conditionnellement à l'état $s_j^t$ et au modèle de HMM $\lambda$ utilisé :

$$\beta_t(s_j^t) = P(O_T, O_{T-1} \ldots O_{t+1}|s_j^t, \lambda), \quad \forall s_j^t \in \Omega_t \tag{5.9}$$

L'estimation de cette probabilité est réalisée par la propagation *backward* qui consiste à combiner les distributions $\beta_{t+1}(s_i^{t+1})$ et $b_i(\mathbf{O}_{t+1})$ disponibles à $t+1$ puis à propager la distribution résultante vers les états $s_j^t \in \Omega_t$ en utilisant les probabilités de transition $a_{ij}$. On obtient donc :

$$\beta_t(s_j^t) \quad = \sum_{\forall s_i^{t+1} \in \Omega_{t+1}} a_{ij} \cdot b_i(\mathbf{O}_{t+1}) \cdot \beta_{t+1}(s_i^{t+1}) \tag{5.10}$$

Le calcul de cette équation est illustré sur la figure 5.3 et une application sur un cas réel pour le *lissage* est représentée sur la figure 5.4c. Remarquons que pour obtenir une distribution, il est nécessaire de normaliser les vraisemblances $\beta_t(s_j^t)$ par $\sum_j \beta_t(s_j^t)$.

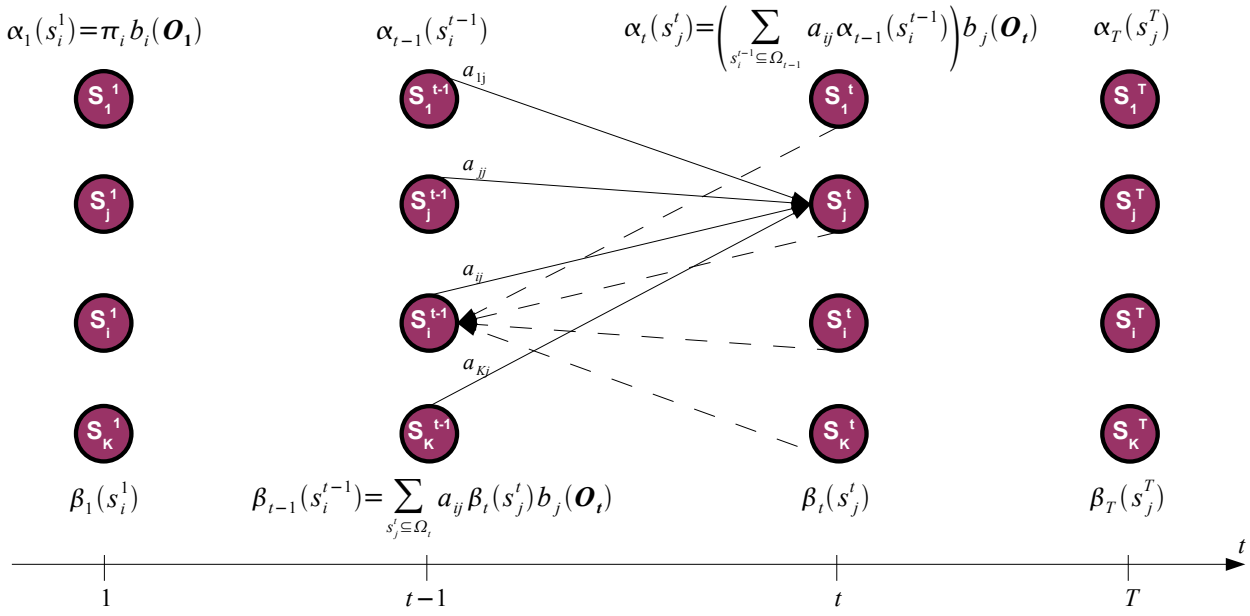



**FIG. 5.3 :** *Propagation des distributions, figure inspirée de [Rab89, MG00]. Les traits continus représentent la propagation* forward*, et les traits pointillés la propagation* backward*. Les expressions des variables du même nom sont rappelées respectivement au-dessus et au-dessous de la figure.*

### 5.2.3 Estimation de la probabilité sur chaque état et filtrage

On cherche à calculer la vraisemblance de chaque état à un instant $t$ tout en assurant une cohérence temporelle entre deux instants consécutifs $t-1$ et $t$. Dans le cas « en ligne », il suffit d'appliquer la propagation *forward*. Dans le cas « hors ligne », on peut appliquer les propagations *forward* et *backward* puis combiner conjonctivement les résultats obtenus à chaque instant. La variable obtenue est généralement notée $\gamma_t$ :

$$\gamma_t(s_j^t) = P(s_j^t|\mathbf{O}_{1:t}, \lambda) \tag{5.11a}$$

$$\propto \alpha_t(s_j^t) \cdot \beta_t(s_j^t) \tag{5.11b}$$

où le coefficient de proportionnalité est donné par $\sum_{j=1}^{K} \alpha_t(s_j^t) \cdot \beta_t(s_j^t)$.

La variable $\gamma_t$ permet d'estimer l'état le plus probable $s_*^t$ à l'instant courant $t$. Pour cela, il suffit de déterminer l'état $s_*^t$ maximisant la vraisemblance sur $\gamma_t$ :

$$s_*^t = \underset{s_j \in \Omega_t}{\operatorname{argmax}}\ \gamma_t(s_j^t) \tag{5.12}$$

Cependant, ce critère de détermination du meilleur état ne prend pas explicitement en compte la séquence des états puisque la maximisation est réalisée instant par instant à la différence de l'algorithme de Viterbi, que nous décrivons dans le paragraphe suivant, qui prend en compte les meilleurs états passés pour déterminer le meilleur état courant.

**Exemple 23.** *On suppose disposer d'un algorithme liant les* 5 *observations de la figure 5.1b aux* 4 *états* courir *($s_1$),* sauter *($s_2$),* chuter *($s_3$) et* se relever *($s_4$) d'un* saut en hauteur. *On pose* $\Omega_t = \{s_1^t, s_2^t, s_3^t, s_4^t\}$.

*L'algorithme liant les observations numériques aux états fournit les vraisemblances* $b_j(\mathbf{O}_t), \forall s_j^t \in \Omega_t, \forall t \in \{1 \dots 153\}$ *(la durée est le nombre d'images composant le saut en hauteur) illustrées sur la figure 5.4a.*

*On cherche à filtrer ces vraisemblances afin d'assurer une cohérence temporelle. La matrice de transition utilisée est donnée dans le tableau 5.1 et les a priori sont fixés à* $P^{\Omega_1}(s_1^1) = P^{\Omega_1}(s_2^1) = P^{\Omega_1}(s_3^1) = P^{\Omega_1}(s_4^1) = 0.25$.

| | $s_1$ | $s_2$ | $s_3$ | $s_4$ |
|---|---|---|---|---|
| $s_1$ | 0.930 | 0.049 | 0.015 | 0.006 |
| $s_2$ | 0.025 | 0.911 | 0.050 | 0.014 |
| $s_3$ | 0.010 | 0.050 | 0.878 | 0.063 |
| $s_4$ | 0.005 | 0.010 | 0.048 | 0.937 |

**TAB. 5.1 :** *Matrice de transition probabiliste* $P^{\Omega_t}(.|\omega_{t-1})$ *obtenue par estimation [Rab89].*

*Les résultats du filtrage par les propagations* forward *(resp.* backward*) probabilistes sont donnés sur la figure 5.4b (resp. Fig. 5.4c). On illustre par ailleurs la combinaison conjonctive des deux distributions à chaque instant (variable* $\gamma_t$*) sur la figure 5.4d.*

### 5.2.4 Recherche de la meilleure séquence d'états

L'algorithme de Viterbi permet de déterminer la séquence d'états $s^1, s^2 \dots s^T$ la plus probable ayant généré la séquence des observations $\mathbf{O}_{1:T}$. Cette séquence optimale est déterminée parmi toutes les séquences possibles ($K^T$ cas pour $K$ états et $T$ observations) par des mécanismes de programmation dynamique.

#### 5.2.4.1 Principe

A chaque instant $t$, on cherche le meilleur état $s_i^{t-1} \in \Omega_{t-1}$ expliquant le fait que l'état en cours est $s_j^t \in \Omega_t$. Pour cela, on détermine l'état $s_i^{t-1}$ à $t-1$ qui donne la vraisemblance maximale $\delta_t(s_j^t)$ pour $s_j^t$ connaissant les transitions $a_{ij}$ et les vraisemblances $b_j(\mathbf{O}_t)$ conditionnelles aux observations. Le principe est illustré sur la figure 5.5.

Formellement, on cherche à maximiser la probabilité $P(s^1, s^2 \dots s_j^t, O_1, O_2 \dots O_t|\lambda)$ d'*une séquence se terminant par l'état* $s_j^t$. Cette probabilité est donnée par :

$$\begin{aligned} \delta_t(s_j^t) &= \max_{s^1, s^2 \dots s^{t-1}} P(s^1, s^2 \dots s_j^t, \mathbf{O}_{1:t}|\lambda) \\ &= \max_{\forall s_i^{t-1} \in \Omega_{t-1}} \left[\delta_{t-1}(s_i^{t-1}) \cdot a_{ij}\right] \cdot b_j(\mathbf{O}_t) \end{aligned} \tag{5.13}$$

La variable $\delta_t$ mémorise à chaque instant $t$ la probabilité maximisant la transition de l'état $s_i^{t-1}$ vers $s_j^t$ ayant observé $b_j(\mathbf{O}_t)$. On peut remarquer que cette phase récursive n'est pas sans rappeler la propagation de *forward* (Eq. 5.8) où l'opérateur « max-produit » remplace l'opérateur « somme-produit ».

L'état $s_i^{t-1}$ le plus probable occupé à l'instant $t-1$ à partir duquel l'état $s_j^t$ a été atteint à $t$ est tel que la quantité $\delta_{t-1}(s_i^{t-1}) \cdot a_{ij} \cdot b_j(\mathbf{O}_t)$ est maximale. On mémorise alors cet état, qui est le *prédécesseur*

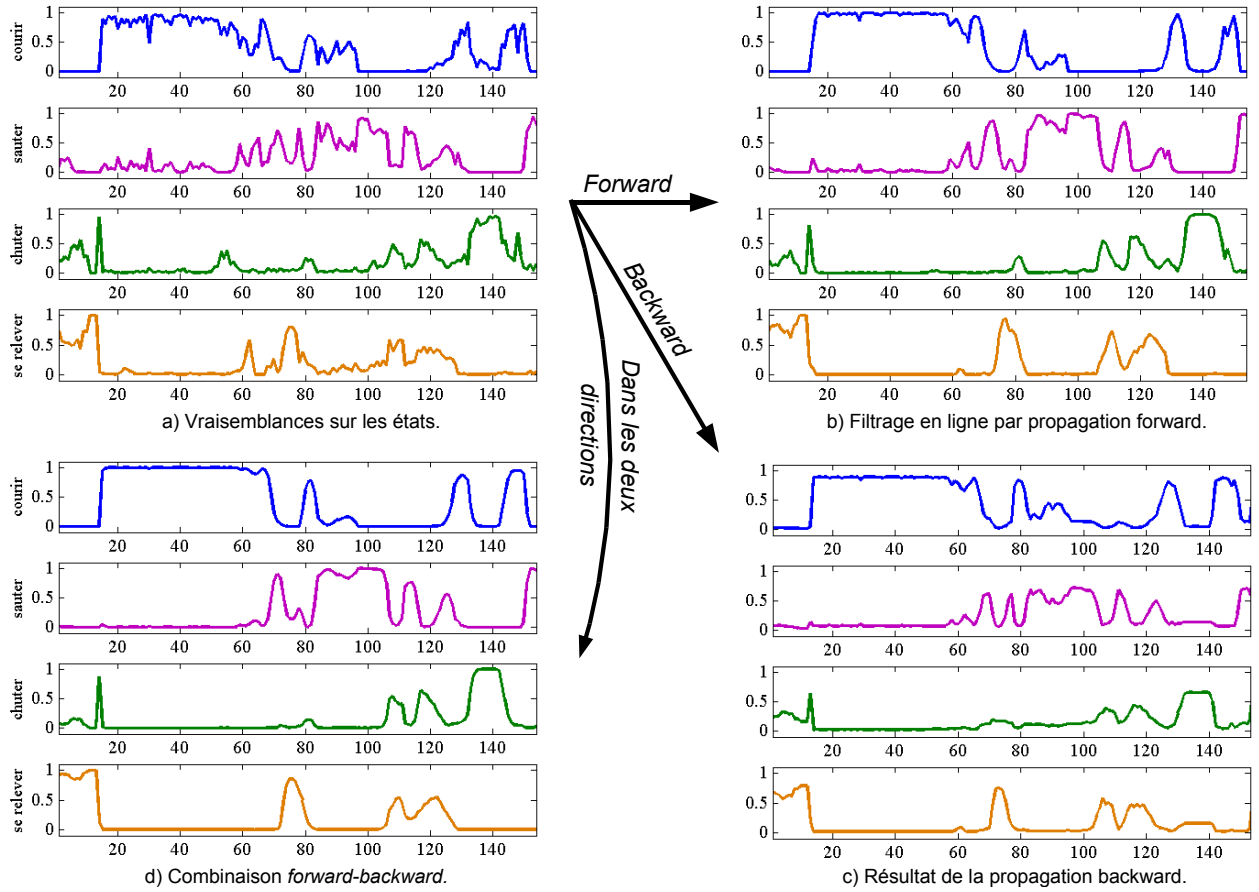



**FIG. 5.4 :** *Résultats du filtrage des (a) vraisemblances (normalisées) sur les états obtenues à l'aide des observations, (b) par la propagation* forward *probabiliste et (c) par la propagation* backward *probabiliste. La figure (d) donne le résultat de la combinaison conjonctive des deux propagations à chaque instant (propagation dans les deux directions).*

de $s_j^t$, dans la variable $\psi_t(s_j^t)$ :

$$\psi_t(s_j^t) = \underset{\forall s_i^{t-1} \in \Omega_{t-1}}{\operatorname{argmax}} \left[\delta_{t-1}(s_i^{t-1}) \cdot a_{ij}\right] \tag{5.14}$$

et donc cette seconde variable $\psi_t(s_j^t)$ est à valeur dans $\Omega_{t-1}$. Lorsque la dernière observation $\mathbf{O}_T$ est atteinte à $T$, l'état le plus probable $s_*^T$ est sélectionné et une procédure de *rétro-propagation* (recherche des prédécesseurs), via la variable $\psi_t$ et partant de $\psi_T(s_*^T)$, est exécutée pour retrouver la meilleure séquence.

#### 5.2.4.2 Etapes de l'algorithme de Viterbi

L'algorithme de Viterbi, dont on a présenté les différents éléments dans les paragraphes précédents, est composé de $4$ étapes :

1. **Initialisation -** A $t = 1$, on ne dispose que des probabilités a priori $\pi_{s_j^1}$ des états $s_j^1$ ainsi que

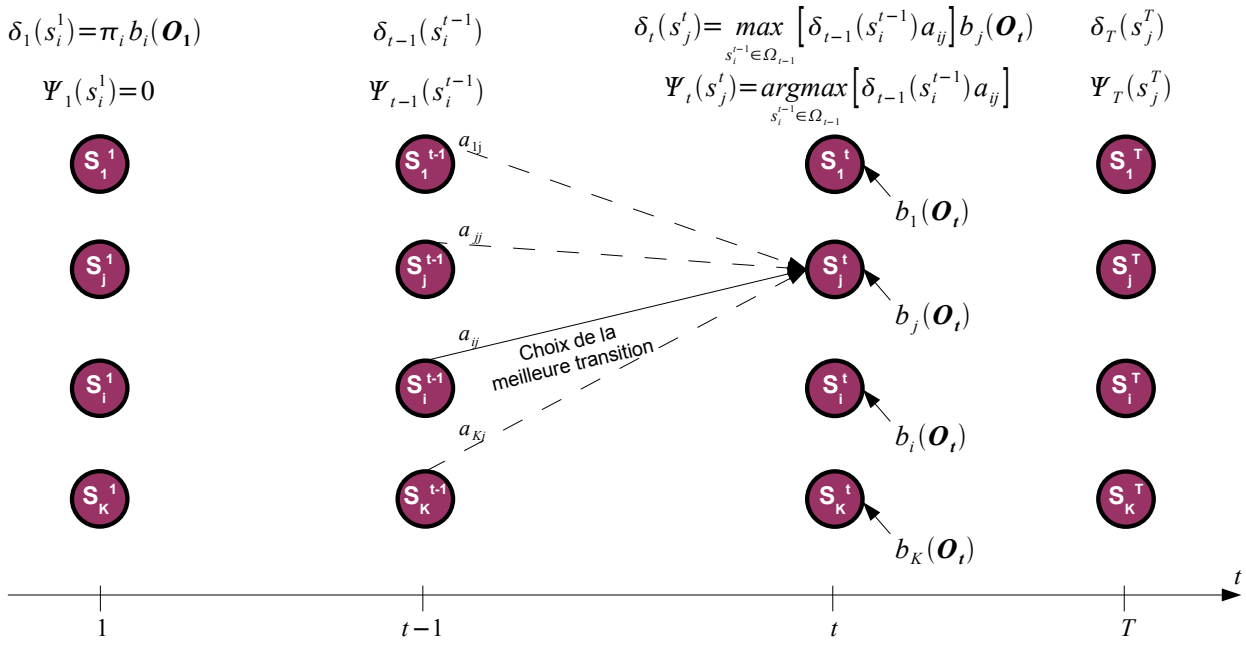



**FIG. 5.5 :** *Algorithme de Viterbi, inspiré de [Rab89, MG00].*

des vraisemblances $b_j(\mathbf{O}_1)$. Il suffit alors de les combiner conjonctivement, $\forall s_j^1 \in \Omega_1$ :

$$\delta_1(s_j^1) = \pi_{s_j^1} \cdot b_j(\mathbf{O}_1) \tag{5.15a}$$

$$\psi_1(s_j^1) = 0 \tag{5.15b}$$

L'initialisation de la variable $\psi_1(s_j^1)$ à 0 indique que $s_j^1$ n'a pas de prédécesseur.

2. **Propagation -** Dans cette phase, on recherche le prédécesseur $s_i^{t-1}$ le plus probable pour chaque état courant $s_j^t$ ainsi que la probabilité générée sur cet état, i.e. $\forall t \in \{2 \dots T\}$, $\forall s_j^t \in \Omega_t$ :

$$\delta_t(s_j^t) = \max_{\forall s_i^{t-1}\in\Omega_{t-1}} \left[\delta_{t-1}(s_i^{t-1}) \cdot a_{ij}\right] \cdot b_j(\mathbf{O}_t)$$

$$\psi_t(s_j^t) = \operatorname*{argmax}_{\forall s_i^{t-1}\in\Omega_{t-1}} \left[\delta_{t-1}(s_i^{t-1}) \cdot a_{ij}\right]$$

3. **Fin -** Lorsque la dernière observation est atteinte à $T$, la probabilité du meilleur état représente la probabilité de la séquence optimale finissant en $s_T^*$ :

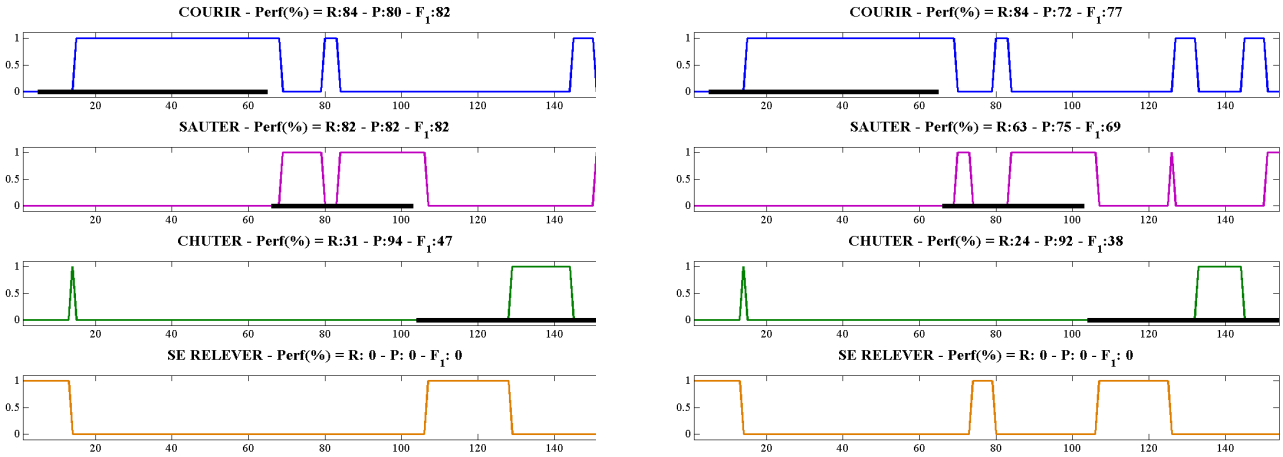
$$P^* = \max_{\forall s_j^T\in\Omega_T} \delta_T(s_j^T) \tag{5.17a}$$

$$s_T^* = \operatorname*{argmax}_{\forall s_j^T\in\Omega_T} \delta_T(s_j^T) \tag{5.17b}$$

4. ***Rétro-propagation* -** Lorsque l'on a déterminé le dernier meilleur état $s_*^T$, il suffit de remonter aux prédécesseurs, i.e. $\forall t \in \{1 \dots T-1\}$, $\forall s^t \in \Omega_t$ :

$$s_t^* = \psi_{t+1}(s_{t+1}^*) \tag{5.18}$$

**Exemple 24.** *On applique l'algorithme de Viterbi sur les vraisemblances de la figure 5.4a pour retrouver la séquence d'états la plus probable. Le résultat est donné sur la figure 5.6a et nous le comparons à la maximisation a posteriori (Eq. 5.12) à chaque instant de la variable $\gamma_t$ combinant les propagations* forward *et* backward*. Les traits gras noirs apparaissant sur l'axe des abcisses représentent la vérité terrain.*




a) Décodage de Viterbi.

b) Maximisation a posteriori sur la combinaison *forward-backward*.

**FIG. 5.6 :** *Résultats de l'algorithme de Viterbi (a) pour la reconnaissance de la séquence d'états la plus probable à partir des observations de la figure 5.4a. Comparaison avec la détermination du meilleur état (b) en utilisant la maximisation a posteriori (Eq. 5.12) sur les probabilités $\gamma_t$ données à la figure 5.4d.*

*Quantitativement, les deux méthodes sont assez proches (l'algorithme de Viterbi est un peu meilleur, voir les résultats des rappels / précisions / $F_1$-mesures sur les figures 5.6a-b). Cependant, l'algorithme de Viterbi nécessite seulement de calculer un* maximum *et de stocker une distribution et un tableau de valeurs alors que le calcul de la variable $\gamma_t$ impose le calcul des variables* forward *et* backward *ainsi que le stockage des résultats de ces propagations puis leur combinaison. De plus, l'algorithme de Viterbi est cohérent avec l'objectif visé à savoir maximiser la probabilité sur la séquence des états alors que la variable $\gamma$ agit instant par instant.*

### 5.2.5 Classification d'une séquence d'observations

Etant donnés une séquence d'observations et $N$ modèles de HMM $\lambda_1, \lambda_2, \ldots, \lambda_n, \ldots, \lambda_N$ représentant chacun un type de séquence particulier (i.e. une *classe*), on cherche à connaître le modèle $\lambda_*$ correspondant le mieux aux observations mesurées $\mathbf{O}_{1:T}$. C'est donc un problème de *classification* de séquences d'observations[3].

Une solution consiste à estimer la probabilité de la séquence des observations $P(\mathbf{O}_{1:T}|\lambda)$ [Rab89] aussi appelée *vraisemblance du modèle* $\lambda$. En remarquant que la variable *forward* (Eq. 5.8) est donnée par $\alpha_t(s_j^t) = P(O_1, O_2 \ldots O_t, s_j^t|\lambda)$, il est possible d'estimer la vraisemblance $P(\mathbf{O}_{1:T}|\lambda)$ en

[3]Ce problème a été traité dans le chapitre 4-§ 4.4 pour l'ordonnanceur CSS.

marginalisant $\alpha_T$ sur les états à l'instant $T$ :

$$\sum_{\forall s_j^T \in \Omega_T} \alpha_T(s_j^T) = \sum_{\forall s_j^T \in \Omega_T} P(O_1, O_2 \dots O_T, s_j^T|\lambda) \tag{5.19}$$

$$= P(O_1, O_2 \dots O_T|\lambda) \tag{5.20}$$

Ainsi, l'application de la propagation *forward* avec les paramètres d'un HMM $\lambda$ nous permet d'estimer la vraisemblance du modèle $\lambda$ conditionnellement aux observations $\mathbf{O}_{1:T}$ mesurées. Le HMM qui maximise la vraisemblance $P(O_1, O_2 \dots O_T|\lambda)$ représente la classe de la séquence d'observations :

$$\lambda^* = \underset{n}{\operatorname{argmax}} \; P(\mathbf{O}_{1:T}|\lambda_n) \tag{5.21}$$

Dans le but de comparer l'approche probabiliste à l'approche crédibiliste que nous verrons par la suite, nous présentons une méthode alternative permettant d'estimer la vraisemblance $P(\mathbf{O}_{1:T}|\lambda_n)$. Pour cela, remarquons que dans l'équation 5.8, estimant $\alpha_t(s_j^t)$, les vraisemblances ne sont pas normalisées à chaque instant. Posons $c_t$ le coefficient de normalisation de $\alpha_t$ tel que :

$$c_t = \frac{1}{\sum_{j=1}^{N} \alpha_t(s_j^t)} \tag{5.22}$$

On note $\alpha_t^*(s_j^t)$ la valeur normalisée (probabilité) de la variable en $s_j^t$ telle que :

$$\alpha_t^*(s_j^t) = c_t \cdot \alpha_t(s_j^t) \tag{5.23}$$

A présent, nous pouvons réécrire la formule de propagation *forward* de la manière suivante :

$$\alpha_t^*(s_j^t) = \left( \sum_{\forall s_i^{t-1} \in \Omega_{t-1}} \alpha_{t-1}^*(s_i^{t-1}) \cdot a_{ij} \right) \cdot b_j(\mathbf{O}_t) \tag{5.24}$$

$$= c_{t-1} \cdot \left( \sum_{\forall s_i^{t-1} \in \Omega_{t-1}} \alpha_{t-1}(s_i^{t-1}) \cdot a_{ij} \right) \cdot b_j(\mathbf{O}_t) \tag{5.25}$$

En réitérant cette opération de $t = 1$ à $t$, on obtient :

$$\alpha_{t-1}^* = \alpha_{t-1} \cdot \prod_{\tau=1}^{t-1} c_\tau \tag{5.26}$$

On obtient donc la relation [Rab89] :

$$\prod_{t=1}^{T} c_t \cdot \sum_{\forall s_j^T \in \Omega_T} \alpha_T(s_j^T) = 1 \tag{5.27}$$

Or, par définition de la variable *forward*, on a $\alpha_T(s_j^T) = P(O_1, O_2 \dots O_T, s_j^T|\lambda)$ et donc la somme sur tous les états à $T$ revient à marginaliser sur les observations comme nous l'avons vu à l'équation 5.20 permettant ainsi d'obtenir la vraisemblance $P(O_1, O_2 \dots O_T|\lambda)$. On obtient donc :

$$P(\mathbf{O}_{1:T}|\lambda) = \prod_{t=1}^{T} \frac{1}{c_t} \tag{5.28}$$

Ainsi, la vraisemblance de la séquence des observations est quantifiée par le produit des inverses des coefficients de normalisation. D'un point de vue implémentation, ce produit est rapidement inférieur aux limites de précision des ordinateurs actuels et on passe généralement par le logarithme pour palier ce problème :

$$\log(P(\mathbf{O}_{1:T}|\lambda)) = -\sum_{t=1}^{T} \log(c_t) \tag{5.29}$$

Disposant de $N$ modèles de HMM $\lambda_1, \lambda_2 \ldots \lambda_n \ldots \lambda_N$ (autant de modèles de séquence que de séquences à reconnaître), on peut déterminer le meilleur modèle ($\lambda_*$) en maximisant le critère a posteriori $\log(P(\mathbf{O}_{1:T}|\lambda_n))$ au lieu de $P(\mathbf{O}_{1:T}|\lambda_n)$ dans l'équation 5.21. Ce type de critère est très utilisé en reconnaissance des formes [BMF04].

## 5.2.6 Apprentissage des paramètres

Les différentes variables et méthodes définies dans les paragraphes précédents sont basées sur les paramètres $a_{ij}$ (transitions), $\pi_i$ (a priori) et $b_j(\mathbf{O}_t)$ (liaison entre observations numériques et vraisemblances sur les états). Dans ce paragraphe, nous présentons des méthodes permettant de les estimer automatiquement à partir d'un ensemble d'apprentissage et les descriptions sont basées sur le tutoriel de Rabiner [Rab89].

### 5.2.6.1 Estimation des modèles d'observation : cas de mélanges Gaussiens

On cherche à lier les observations $\mathbf{O}_{1:T}$ numériques aux états symboliques. On rappelle que les états sont cachés et que seulement $F$ observations $\mathbf{O}_t = [O_1, O_2 \ldots O_F]$, où $\mathbf{O}_t \in \Re^F$, sont mesurées à un instant $t$, les observations reflétant l'état du système dynamique à étudier (un exemple est donné sur la figure 5.1a-b).

**Mélanges Gaussiens** Classiquement, le lien entre les observations $\mathbf{O}_t$, généralement bruitées, et les $K$ états dans $\Omega_t$ d'un HMM $\lambda$ correspond à une transformation dite « numérique-symbolique ». Celle-ci prend fréquemment la forme d'un mélange Gaussien qui permet de prendre en compte les liens complexes entre les observations.

Un mélange Gaussien est défini par une somme pondérée de Gaussiennes :

$$b_j(\mathbf{O}_t) = P(\mathbf{O}_t|s_j^t) = \sum_{g=1}^{G} C_{j,g} \cdot \mathcal{N}(\mathbf{O}_t; \mu_{j,g}, \Sigma_{j,g}) \tag{5.30a}$$

$$= \sum_{g=1}^{G} C_{j,g} \cdot \frac{1}{2\pi^{\frac{F}{2}} |\Sigma|^{\frac{1}{2}}} \cdot \exp((\mathbf{O}_t - \mu)' \Sigma^{-1} (\mathbf{O}_t - \mu)) \tag{5.30b}$$

où $\mathcal{N}$ est une Gaussienne de moyenne $\mu$ et de variance $\Sigma$ et où $C_{j,g} = P(g|s_j^t)$ est le poids de la Gaussienne $g$ de l'état $s_j^t$. A une observation $\mathbf{O}_t$ correspond une valeur $b_j(\mathbf{O}_t)$ qui correspond à la *vraisemblance de l'observation conditionnellement à l'état* $s_j^t$. Par la suite on parlera de *vraisemblance de l'observation* par simplification.

Le problème de l'apprentissage des modèles d'observation revient donc à estimer les moyennes, variances et pondérations des mélanges Gaussiens. Nous rappelons dans les paragraphes suivants les principaux résultats (voir la référence de Rabiner [Rab89] pour plus de détails).

**Estimation des paramètres des Gaussiennes** La variable $\gamma$ définie à l'équation 5.11b à partir des propagations *forward* et *backward* permet d'estimer la moyenne ($\overline{\boldsymbol{\mu}}$), la variance ($\overline{\boldsymbol{\Sigma}}$) et la pondération ($\overline{C}$) des Gaussiennes des mélanges Gaussiens sur un état $s_j^t$. Les équations d'estimation sont basées sur une version conditionnée de la variable $\gamma$ à savoir [Rab89] :

$$\gamma_t^*(s_j^t, g) = \gamma_t(s_j^t) \cdot \frac{P(\mathbf{O}_t|s_j^t, g)}{P(\mathbf{O}_t|s_j^t)} \tag{5.31}$$

où $P(\mathbf{O}_t|s_j^t, g)$ est obtenue de la façon suivante :

$$P(\mathbf{O}_t|s_j^t, g) \quad = \quad \mathcal{N}(\mathbf{O}_t; \mu_{j,g}, \Sigma_{j,g}) \tag{5.32}$$

La variable $\gamma_t^*(s_j^t, g)$ représente une version « détaillée » de la variable $\gamma_t(s_j^t)$ : on retire tout d'abord la quantité $P(\mathbf{O}_t|s_j^t)$ à $\gamma_t(s_j^t)$ car elle est prise en compte dans le calcul de la variable *forward*, puis le résultat obtenu est pondéré par $P(\mathbf{O}_t|s_j^t, g)$ représentant la probabilité des observations sur la composante $g$ du mélange Gaussien attaché à l'état $s_j^t$.

Les équations d'estimation sont alors données par [Rab89] :

$$\overline{C}_{j,g} \propto \sum_{t=1}^{T} \gamma_t^*(s_j^t, g) \tag{5.33a}$$

$$\overline{\boldsymbol{\mu}}_{j,g} \propto \sum_{t=1}^{T} \gamma_t^*(s_j^t, g) \cdot \mathbf{O}_t \tag{5.33b}$$

$$\overline{\boldsymbol{\Sigma}}_{j,g} \propto \sum_{t=1}^{T} \gamma_t^*(s_j^t, g) \cdot (\mathbf{O}_t - \boldsymbol{\mu}_{j,g})(\mathbf{O}_t - \boldsymbol{\mu}_{j,g})' \tag{5.33c}$$

où $(\mathbf{O}_t - \boldsymbol{\mu}_{j,g})(\mathbf{O}_t - \boldsymbol{\mu}_{j,g})'$ est une multiplication matricielle (« $'$ » est une transposition), $P(\mathbf{O}_t|s_j^t)$ est donnée par l'équation 5.30. On rappelle que $\propto$ signifie « proportionnel à » et les coefficients de proportionnalité sont ici obtenus en sommant chacun des paramètres sur $j \in \{1 \dots K\}$ et sur $g \in \{1 \dots G\}$.

On remarque que l'estimation des paramètres fait appel à ces mêmes paramètres et c'est pourquoi le processus d'apprentissage est généralement itératif [Rab89, Mur02].

#### 5.2.6.2 Estimation de l'a priori

La variable $\gamma$ définie à l'équation 5.11b permet d'estimer la distribution a priori sur les états en prenant la distribution $\gamma_1$ en $t = 1$ [Rab89] :

$$\overline{\Pi} = \gamma_1 \tag{5.34}$$

Il est bien entendu possible de fixer manuellement la distribution a priori si celle-ci est connue.

#### 5.2.6.3 Estimation de la matrice de transition

La matrice de transition représente l'ensemble des liens statistiques entre les états à des instants consécutifs. Pour l'estimer, on définit la variable $\xi_t(s_i^t, s_j^{t+1}) = P(s_i^t, s_j^{t+1}|\mathbf{O}_{1:T}, \lambda)$ qui représente la probabilité d'occurrence de deux états $s_i^t$ et $s_j^{t+1}$ à deux instants successifs et conditionnellement au modèle et aux observations. Cette variable peut être calculée par l'expression suivante :

$$\xi_t(s_i^t, s_j^{t+1}) \propto \alpha_t(s_i^t) \cdot a_{ij} \cdot b_j(\mathbf{O}_{t+1}) \cdot \beta_{t+1}(s_j^{t+1}) \tag{5.35}$$

où le coefficient de proportionnalité est donné par $\sum_{i=1}^{K}\sum_{j=1}^{K}\alpha_t(s_i^t)\cdot a_{ij}\cdot b_j(\mathbf{O}_{t+1})\cdot\beta_{t+1}(s_j^{t+1})$. Le calcul de la variable $\xi_t$ met en jeu les variables $\alpha_t(s_i^t)$ et $\beta_{t+1}(s_j^{t+1})$ aux instants $t$ et $t+1$ respectivement tandis que le coefficient $a_{ij}\cdot b_j(\mathbf{O}_{t+1})$ permet de projeter $\beta_{t+1}(s_j^{t+1})$ vers $\Omega_t$ afin de relier statistiquement les deux états $s_i^t$ et $s_j^{t+1}$.

On peut remarquer que la marginalisation de $\xi_t$ sur $\Omega_t$ permet d'obtenir la variable $\gamma_t(s_i^t)$ :

$$\sum_{s_j^t\in\Omega_{t+1}}\xi_t(s_i^t,s_j^{t+1})=\gamma_t(s_i^t) \tag{5.36}$$

La variable $\xi_t$ normalisée est utilisée pour estimer la matrice de transition d'un HMM par la moyenne des $\xi$ sur l'ensemble de la séquence [Rab89] :

$$\overline{a}_{ij}\propto\sum_{t=1}^{T-1}\xi_t(s_i^t,s_j^{t+1}) \tag{5.37}$$

où le coefficient de proportionnalité est donné par $\sum_{t=1}^{T-1}\sum_{s_j^t\in\Omega_{t+1}}\xi_t(s_i^t,s_j^{t+1})=\sum_{t=1}^{T-1}\gamma_t(s_i^t)$ qui assure que la matrice de transition est stochastique.

Cette équation est généralement intégrée dans un processus itératif tel que l'EM (*Espérance-Maximisation*) [Rab89] puisque la variable $\xi_t$ utilisée dans l'estimation de $\overline{a}_{ij}$ fait justement appel aux transitions $a_{ij}$.

Les principaux mécanismes des HMM sont résumés dans le tableau 5.2.

| | |
|---|---|
| Filtrage d'une distribution temporelle | Eq. 5.8 |
| Classification de séquences | Eq. 5.21 |
| Séquence d'états optimale | § 5.2.4.2 |
| Estimation de l'a priori | Eq. 5.34 |
| Estimation des modèles | Eq. 5.33 |
| Estimation des transitions | Eq. 5.37 |

**TAB. 5.2 :** *Résumé des principaux mécanismes des HMM.*

### 5.2.7 Pourquoi étendre les HMM aux fonctions de croyance ?

Les HMM probabilistes offrent de très bonnes performances mais des questions persistent auxquelles la théorie des probabilités répond difficilement :

- Comment utiliser le HMM si les transformations numérique-symbolique des observations en états ne sont pas toutes des distributions de probabilités, ou si les transitions entre états ne sont pas toutes des probabilités conditionnelles ? Le TBM offre un cadre très général pour combiner des informations de types différents.

- Comment combiner des distributions non indépendantes ? Des solutions existent dans le cadre probabiliste mais impose de connaître le type de dépendance afin de pouvoir l'estimer et le corriger. Dans le cadre du TBM, les BBA issues de sources dépendantes sont combinées avec l'opérateur prudent (§ 2.2.5).

– Que représente la normalisation opérée lors des combinaisons ? Dans le cadre du TBM, la normalisation est liée au conflit et nous exploitons cette information pour la reconnaissance de séquences.

– Que se passe t-il si la distribution sur les états initiaux est inconnue ? La solution employée dans le cadre de la théorie des probabilités est de faire appel au principe d'*équiprobabilité* consistant à affecter la même probabilité à chaque état. Dans le cadre des fonctions de croyance, cette *méconnaissance* est modélisée par la fonction vide.

– Comment traduire l'ambiguïté sur les transitions entre deux états à un même instant ($s_i^t$ et $s_j^t$) ou entre deux instants consécutifs ($s_i^t$ et $s_j^{t+1}$) ? Les méthodes probabilistes exploite là encore le principe d'équiprobabilité alors que dans le cadre du TBM, une ambiguité est directement modélisée par une fonction de croyance (par exemple $m^{\Omega_t}(\{s_i, s_j\}) = 1$).

Nous proposons de formaliser les algorithmes des HMM dans le cadre du TBM afin de combiner la puissance des algorithmes des HMM, la flexibilité et les outils tout aussi puissants du TBM et enfin de profiter de la généricité des fonctions de croyance qui permettent de modéliser un large panel de connaissances. Notre objectif est d'offrir un outil générique permettant d'utiliser les connaissances telles qu'elles existent, que ce soit des probabilités, des possibilités ou des fonctions de croyance.

## 5.3 HMM crédibiliste

L'idée d'étendre les HMM à la théorie de l'évidence [Sha76] a été proposée par Pieczynski *et al.* (voir [Pie07] pour une référence récente) dans des applications liées au traitement d'image. Dans ces travaux, l'objectif est d'exploiter les fonctions de croyances afin de s'affranchir de l'a priori nécessaire dans le calcul de distributions a posteriori avec la relation de Bayes. Le problème des méthodes proposées est que les observations ou bien les a priori sont des fonctions de masse mais pas les deux simultanément. En conséquence, la combinaison donne forcément une distribution de probabilités. De plus, les distributions de croyances sont généralement obtenues par simple affaiblissement (§ 2.1.4.4) de distributions de probabilités [LP05a, LP05b] afin de transférer de la croyance sur le doute total. Enfin, dans les méthodes proposées par Pieczynski *et al.*, la règle de Dempster (Eq. 2.10) est utilisée sans prêter attention à la normalisation (§ 2.1.6 et réf. [Sme94a, Sme92]).

Dans ce chapitre, les algorithmes proposés sont applicables à des fonctions de croyances quelconques. De plus, nous nous plaçons dans le cadre du TBM de Smets et non pas la théorie de Dempster-Shafer [Sha76] (les différences entre les deux modèles ont été discutées dans [Sme94a]). Les nombreux outils et le formalisme puissant du TBM (GBT, opérateurs de changement d'espace, prise en compte du conflit explicite, opérateurs de combinaison, prise de décision ...) confèrent à ce modèle de nombreux avantages par rapport aux autres modèles basés sur les fonctions de croyances.

En particulier, le conflit est largement exploité dans les algorithmes que nous proposons alors que, dans les méthodes proposées dans la littérature, le conflit est généralement éliminé (par exemple en normalisant). Par rapport aux travaux précédents, nous proposons par ailleurs un algorithme de Viterbi crédibiliste.

La puissance des HMM tient dans les mécanismes de propagation *forward* et *backward* utilisés largement dans les outils des HMM. Ces mécanismes ont leur équivalent dans le cadre du TBM et par conséquent il est possible d'adapter les algorithmes des HMM pour traiter des fonctions de croyance. Dans les paragraphes suivants, nous décrivons les algorithmes de filtrage, reconnaissance et apprentissage dans le cadre des fonctions de croyance et du Modèle des Croyances Transférables en suivant le même plan que pour la description des HMM probabilistes.

### 5.3.1 Introduction

On rappelle que l'ensemble des états est noté $\Omega_t = \{s_1^t, s_2^t \dots s_K^t\}$ et qu'une distribution de croyances peut prendre la forme, entre autres (§ 2.1.2.2), d'une distribution de masses (BBA, $m$), de plausibilités ($pl$) et de communalités ($q$). Nous notons ces fonctions par $f^{\Omega_t} = \{m^{\Omega_t}, pl^{\Omega_t}, q^{\Omega_t}\}$, celles-ci étant en relation biunivoque.

Les extensions des algorithmes des HMM aux fonctions de croyance fait intervenir principalement les communalités qui permettent d'obtenir, dans la forme, des relations proches des expressions probabilistes.

| Nom des algorithmes et variables | Proba. | Equation | TBM | Equation |
|---|---|---|---|---|
| Observations | $b_j(\mathbf{O}_t)$ | Eq. 5.30 | $f_b^{\Omega_t}$ | - |
| Transitions | $a_{ij}$ | Eq. 5.3 | $f_a^{\Omega_t}[S_i^{t-1}]$ | - |
| A priori | $\Pi$ | Eq. 5.34 | $f^{\Omega_0}$ | - |
| Variable *forward* | $\alpha$ | Eq. 5.8 | $f_\alpha^{\Omega_t}$ | 5.41 |
| Variable *backward* | $\beta$ | Eq. 5.10 | $f_\beta^{\Omega_t}$ | 5.45 |
| Variable $\gamma$ | $\gamma$ | Eq. 5.11b | $f_\gamma^{\Omega_t}$ | 5.46 |
| Métrique de Viterbi | $\delta$ | Eq. 5.13 | $f_\delta^{\Omega_t}$ | 5.58 |
| Variable $\xi$ | $\xi$ | Eq. 5.35 | $f_\xi^{\Omega_t}$ | 5.78b |

**TAB. 5.3 :** *Notations probabilistes (colonne « Proba. ») et crédibilistes (colonne « TBM ») des différentes variables et algorithmes des HMM avec les équations correspondantes.*

Notre démarche pour adapter les mécanismes aux fonctions de croyance consiste à transformer les mécanismes de base des HMM probabilistes afin de pouvoir gérer les fonctions de croyance. Le tableau 5.3 résume les notations utilisées et rappelle les numéros d'équations. Par exemple, $f_\alpha^{\Omega_t}$ (ligne 5) est la fonction de croyances équivalente à la variable *forward* probabiliste $\alpha_t$ tandis que $f_a^{\Omega_t}[S_i^{t-1}]$ (ligne 3) représente la fonction de croyance définie sur $\Omega_t$ conditionnellement à $S_i^{t-1} \subseteq \Omega_{t-1}$. L'utilisation de majuscules est réservée aux sous-ensembles, par exemple $S_i^{t-1} \subseteq \Omega_{t-1}$, et les minuscules aux hypothèses, par exemple $s_i^{t-1} \in \Omega_{t-1}$.

### 5.3.2 Mécanismes de base

Les propagations *forward* et *backward* crédibilistes ont été décrits dans le chapitre 2 § 2.2.1.3 et représentent les mécanismes de base des HMM crédibilistes. Les propagations respectent les mêmes schémas que ceux présentés sur la figure 5.2 dans le cas probabiliste et nous les avons illustrés de nouveau sur la figure 5.7 avec les notations crédibilistes. On reprend dans ce paragraphe ces méca-

nismes de base en explicitant la notation temporelle et en utilisant les communalités afin de faire un parallèle avec les probabilités.

**FIG. 5.7 :** *Modèle graphique représentant les propagations* forward *et* backward *crédibilistes entre deux instants où $\Omega_t$ est l'ensemble des états.*

#### 5.3.2.1 Propagation *forward* crédibiliste

Pour décrire la propagation *forward* crédibiliste (Fig. 5.7a), nous avons besoin des éléments suivants :

- $m_\alpha^{\Omega_{t-1}}$ la BBA de la variable *forward* sur les états et sous-ensembles d'états à $t-1$,
- $q_a^{\Omega_t}[S_i^{t-1}], \forall S_i^{t-1} \subseteq \Omega_{t-1}$ l'ensemble des distributions de communalités conditionnelles liant les états et sous-ensembles d'états entre $t-1$ et $t$,
- $q_b^{\Omega_t}$ les communalités obtenues à partir des observations courantes (à $t$).

Comme dans le cas probabiliste, la propagation *forward* respecte un schéma « prédiction – mise à jour » :

- **Etape 1 : calcul de la prédiction $\hat{q}_\alpha^{\Omega_t}(S_j^t)$.** La prédiction de la communalité de la proposition $S_j^t$ est obtenue par application de l'équation 2.28 définie dans le cadre du Théorème de Bayes Généralisé :

$$\hat{q}_\alpha^{\Omega_t}(S_j^t) \quad = \sum_{S_i^{t-1} \subseteq \Omega_{t-1}} m_\alpha^{\Omega_{t-1}}(S_i^{t-1}) \cdot q_a^{\Omega_t}[S_i^{t-1}](S_j^t), \quad \forall S_j^t \subseteq \Omega_t \tag{5.38}$$

  Cette relation estime la somme des contributions des éléments focaux $S_i^{t-1}$ de l'instant précédent pondérées par la transition de $S_i^{t-1}$ vers $S_j^t$. Généralement, seules les croyances conditionnelles aux singletons $s_i^{t-1} \in \Omega_{t-1}$ sont disponibles, la croyance conditionnelle aux unions d'hypothèses peut être calculée en utilisant un opérateur disjonctif[4].

- **Etape 2 : prise en compte des observations.** La prédiction est ensuite combinée conjonctivement avec les observations pour mettre à jour la distribution de croyances sur les états. Dans le cadre crédibiliste, la combinaison conjonctive utilisée est fonction de la dépendance entre $q_\alpha^{\Omega_t}$ et les $q_b^{\Omega_t}$ :
  - Si les sources de croyances sont distinctes alors nous utilisons l'opérateur CRC :

$$q_\alpha^{\Omega_t}(S_j^t) \quad = \quad \hat{q}_\alpha^{\Omega_t}(S_j^t) \cdot q_b^{\Omega_t}(S_j^t), \quad \forall S_j^t \subseteq \Omega_t \tag{5.39}$$

[4]L'opérateur disjonctif peut être soit l'équation 2.29 (règle de combinaison disjonctive DRC) si les croyances sont distinctes ou bien l'équation 2.65 (règle de combinaison disjonctive *hardie* BDRC) si les croyances ne sont pas distinctes.

- Si les sources de croyances ne sont pas distinctes alors nous utilisons l'opérateur prudent CCRC (en transformant les communalités en BBA avec l'Eq. 2.7b) :

$$m_\alpha^{\Omega_t} = \hat{m}_\alpha^{\Omega_t} \circledwedge m_b^{\Omega_t} \tag{5.40}$$

Dans le cas où les sources sont distinctes, la distribution sur les états à $t$ après les phases de prédiction et de mise à jour est donnée par :

$$\boldsymbol{q_\alpha^{\Omega_t}(S_j^t) = \Big( \sum_{S_i^{t-1} \subseteq \Omega_{t-1}} m_\alpha^{\Omega_{t-1}}(S_i^{t-1}) \cdot q_a^{\Omega_t}[S_i^{t-1}](S_j^t) \Big) \cdot q_b^{\Omega_t}(S_j^t), \quad \forall S_j^t \subseteq \Omega_t} \tag{5.41}$$

Cette équation est proche, dans la forme, de son homologue probabiliste (Eq. 5.8) puisque seules des sommations et des produits interviennent et ceci grâce aux communalités. Cependant, la version crédibiliste fait intervenir non seulement les hypothèses mais aussi les sous-ensembles d'hypothèses. Elle généralise l'équation 5.8 lorsque les distributions utilisées sont des probabilités : le Théorème de Bayes Généralisé revient au Théorème de Bayes [Sme93] et la combinaison conjonctive (avec normalisation par l'équation 2.10) à un produit de probabilités. De plus, un des avantages de la version crédibiliste par rapport aux probabilités est la possibilité de combiner des croyances fournies par des sources non distinctes sans modéliser la dépendance.

Notons par ailleurs que le conflit peut être non nul car l'équation 5.8 est définie par une double conjonction. La valeur du conflit est donnée par :

$$m_\alpha^{\Omega_t}(\emptyset) = 1 + \sum_{S_j^t \subseteq \Omega_t, S_j^t \neq \emptyset} (-1)^{|S_j^t|} \cdot q_\alpha^{\Omega_t}(S_j^t) \tag{5.42}$$

où $|S_j^t|$ est la cardinalité de $S_j^t$.

Le conflit doit être géré à chaque instant car sinon il est serait réinjecté à l'itération suivante (l'équation 5.8 étant récursive) or, ce conflit est absorbant par la conjonction. Dans la majorité des tests réalisés dans ce manuscrit, nous avons utilisé soit la normalisation soit la règle de Dubois et Prade (voir détails au paragraphe 2.1.6). Nous discuterons de la gestion du conflit dans la section 5.3.5.

**Remarque 5.1.** *La complexité de l'algorithme* forward *probabiliste est en $\mathcal{O}(N^2T)$, elle est de l'ordre de $\mathcal{O}(2^{2N}T)$ (dans le pire des cas) en utilisant les communalités car la distribution de croyances est définie sur l'ensemble des parties des valeurs possibles de la variable. Il existe des méthodes permettant de réduire cette complexité basées soit sur des approximations de cadres de discernement [DY02] soit en passant par les plausibilités [Sme93] (la complexité est alors en $\mathcal{O}(N2^NT)$). De plus, il est à noter que la valeur des masses des éléments focaux peut être très faible dans le cas où $N$ est grand ($N > 12$). Dans ce cas, il est possible de seuiller les valeurs des masses de croyance pour ne conserver que celles qui ont une valeur « importante ». D'un point de vue pratique, lorsque $N$ est grand, il est rare que tous les ensembles d'hypothèses de l'espace puissance soit un élément focal et donc il est préférable de ne pas utiliser les formules matricielles du TBM [Sme02a] mais plutôt opter pour le codage des informations utiles en utilisant le format binaire [HL03].*

#### 5.3.2.2 Propagation *backward* crédibiliste

Pour décrire la propagation *backward* crédibiliste (Fig. 5.7b), nous avons besoin des éléments suivants :

- $m_\beta^{\Omega_{t+1}}$ la BBA de la variable *backward* sur les états et sous-ensembles d'états à $t+1$,
- $pl_a^{\Omega_{t+1}}[S_i^t]$, $\forall S_i^t \subseteq \Omega_t$ l'ensemble des distributions de plausibilités liant les états et sous-ensembles d'états entre $t$ et $t+1$,
- $m_b^{\Omega_{t+1}}$ les BBA obtenues à partir des observations à $t+1$.

A partir de ces fonctions de croyance, il est possible d'estimer la distribution de communalités sur les états à $t+1$ en combinant conjonctivement les distributions à $t+1$ puis en projetant la distribution résultante sur $\Omega_t$ :

- **Etape 1 : calcul de l'a priori en $\Omega_{t+1}$.** En $t+1$, on dispose des BBA $m_\beta^{\Omega_{t+1}}$ et $m_b^{\Omega_{t+1}}$ qu'il suffit de fusionner avec une combinaison conjonctive. Dans le cas de sources distinctes, on obtient :

$$m_{\beta \textcircled{\cap} b}^{\Omega_{t+1}} = m_\beta^{\Omega_{t+1}} \textcircled{\cap} m_b^{\Omega_{t+1}} \tag{5.43}$$

- **Etape 2 : projection vers $\Omega_t$.** La BBA $m_{\beta \textcircled{\cap} b}^{\Omega_{t+1}}$ est tout d'abord propagée vers les états de $\Omega_t$ en calculant la communalité a posteriori $q_a^{\Omega_t}[S_j^{t+1}]$ à partir des plausibilités $pl_a^{\Omega_{t+1}}[S_i^t]$ et de l'équation 2.24b :

$$q_a^{\Omega_t}[S_j^{t+1}](S_i^t) = \prod_{s_i^t \in S_i^t} pl^{\Omega_{t+1}}[s_i^t](S_j^{t+1}) \tag{5.44}$$

Puis, on applique la combinaison conjonctive introduite dans le contexte du GBT (Eq. 2.28) afin d'obtenir le résultat de la propagation *backward* :

$$\boldsymbol{q_\beta^{\Omega_t}(S_i^t) = \sum_{S_j^{t+1} \subseteq \Omega_{t+1}} \left( m_{\beta \textcircled{\cap} b}^{\Omega_{t+1}}(S_j^{t+1}) \cdot \prod_{s_i^t \in S_i^t} pl^{\Omega_{t+1}}[s_i^t](S_j^{t+1}) \right)} \tag{5.45}$$

où le conditionnement est réalisé sur les singletons $s_i^t \in \Omega_t$.

Au départ de la propagation *backward* (en $t = T$), la variable du même nom est initialisée avec la fonction vide modélisant ainsi l'ignorance sur les états de départ, i.e. $q_\beta^{\Omega_T}(S_i^T) = 1, \forall S_i^T \subseteq \Omega_T$.

### 5.3.3 Croyances sur chaque état et filtrage

On cherche à calculer la croyance sur chaque état à un instant $t$ tout en assurant une cohérence temporelle entre deux instants consécutifs $t-1$ et $t$ comme pour l'approche probabiliste. Dans le cas « en ligne », il suffit d'appliquer la propagation *forward*. Dans le cas « hors ligne », l'ensemble des observations de $t = 1$ à $T$ est disponible. Il est donc possible de combiner conjonctivement à chaque instant $t$ les résultats des variables *forward* et *backward* pour prendre en compte toutes les informations disponibles et permettre d'estimer la distribution sur l'ensemble des états et sous-ensembles d'états $S_j^t \subseteq \Omega_t$. On obtient alors l'équivalent crédibiliste de la variable $\gamma$ et que l'on note $f_\gamma^{\Omega_t}$. Pour les communalités ($f \equiv q$), on a :

$$q_\gamma^{\Omega_t}(S_j^t) = q_\alpha^{\Omega_t}(S_j^t) \cdot q_\beta^{\Omega_t}(S_j^t), \quad \forall S_j^t \subseteq \Omega_t \tag{5.46}$$

La variable crédibiliste $q_\gamma^{\Omega_t}$ peut être par ailleurs utilisée pour déterminer le « meilleur » état courant $s_t^*$ à $t$ conditionnellement aux observations. Pour cela, il est nécessaire de calculer la probabilité pignistique sur les états $\Omega_t$ puis de chercher l'état maximisant cette probabilité :

$$s_t^* = \underset{s_i \in \Omega_t}{\operatorname{argmax}} \ \mathbf{BetP}\{m_\gamma^{\Omega_t}\}(s_i), \quad \forall t \in \{1 \dots T\} \tag{5.47}$$

où $\mathbf{BetP}$ est la probabilité pignistique donnée par l'équation 2.16 et $m_\gamma^{\Omega_t}$ est obtenue à partir de $q_\gamma^{\Omega_t}$ en appliquant l'équation 2.7b.

**Exemple 25.** *On reprend l'exemple 23 concernant la reconnaissance d'une séquence de saut en hauteur composée de* $4$ *états :* courir *(*$s_1$*),* sauter *(*$s_2$*),* chuter *(*$s_3$*) et* se relever *(*$s_4$*), et on note* $\Omega_t = \{s_1^t, s_2^t, s_3^t, s_4^t\}$.

*Les fonctions de croyance sur* $\Omega_t$ *sont obtenues à partir des vraisemblances* $b_j(\mathbf{O}_t), \forall s_j^t \in \Omega_t, \forall t \in \{1 \dots 153\}$ *de l'exemple 23 en deux étapes [DS04] : nous les interprétons comme des plausibilités conditionnelles* $pl_b^{\Re^F}[s_j^t](\mathbf{O}_t) \equiv b_j(\mathbf{O}_t)$*, puis nous appliquons le GBT générant les plausibilités* $pl_b^{\Omega_t}[\mathbf{O}_t](S_j^t)$ *qui sont ensuite transformées en probabilités pignistiques. Ces pignistiques sont présentées sur la figure 5.8a. Par la suite, et pour simplifier l'écriture, on omettra le conditionnement* $[\mathbf{O}_t]$ *et on notera* $pl_b^{\Omega_t}(S_j^t)$ *la plausibilité des états conditionnelles aux observations.*

*On cherche à filtrer les croyances* $f_b^{\Omega_t}(S_j^t)$ *sur les états (et donc indirectement les pignistiques associées) afin d'assurer une cohérence temporelle. La matrice de transition utilisée est donnée dans le tableau 5.4 et la communalité initiale est supposée vide (*$q_\alpha^{\Omega_1}(S_i^1) = 1, \forall S_i^1 \subseteq \Omega_1$*). La matrice de transition crédibiliste utilisée pour l'application des propagations* forward *et* backward *est donnée dans le tableau 5.4 (la transformée pignistique redonne la matrice de transition 5.1 utilisée dans le cas probabiliste).*

| | $\emptyset_j$ | $\{s_1^t\}$ | $\{s_2^t\}$ | $\{s_1^t, s_2^t\}$ | $\{s_3^t\}$ | $\{s_1^t, s_3^t\}$ | $\{s_2^t, s_3^t\}$ | $\{s_1^t, s_2^t, s_3^t\}$ | $\{s_4^t\}$ | $\dots$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $[s_1^{t-1}]$ | 0 | 0.889 | 0.008 | 0.059 | 0.002 | 0.001 | 0.003 | 0.018 | 0.001 | $\dots$ |
| $[s_2^{t-1}]$ | 0 | 0.008 | 0.866 | 0.004 | 0.006 | 0.001 | 0.053 | 0.020 | 0.021 | $\dots$ |
| $[s_3^{t-1}]$ | 0 | 0.002 | 0.009 | 0.001 | 0.829 | 0.001 | 0.010 | 0.002 | 0.010 | $\dots$ |
| $[s_4^{t-1}]$ | 0 | 0.001 | 0.002 | 0.001 | 0.006 | 0.001 | 0.001 | 0.001 | 0.894 | $\dots$ |

| $\dots$ | $\{s_1^t, s_4^t\}$ | $\{s_2^t, s_4^t\}$ | $\{s_1^t, s_2^t, s_4^t\}$ | $\{s_3^t, s_4^t\}$ | $\{s_1^t, s_3^t, s_4^t\}$ | $\{s_2^t, s_3^t, s_4^t\}$ | $\Omega_t$ |
|---|---|---|---|---|---|---|---|
| $\dots$ | 0.001 | 0.001 | 0.001 | 0.001 | 0.001 | 0.001 | 0.017 |
| $\dots$ | 0.001 | 0.002 | 0.001 | 0.003 | 0.001 | 0.005 | 0.031 |
| $\dots$ | 0.001 | 0.001 | 0.001 | 0.038 | 0.001 | 0.084 | 0.025 |
| $\dots$ | 0.001 | 0.001 | 0.001 | 0.069 | 0.001 | 0.011 | 0.014 |

**TAB. 5.4 :** *Matrice de transition du CrHMM.*

*Les résultats du filtrage par les propagations* forward *(resp.* backward*) crédibilistes sont donnés sur la figure 5.8b (resp. Fig. 5.8c). On illustre par ailleurs la combinaison conjonctive des deux distributions à chaque instant (variable* $q_\gamma^{\Omega_t}$*) sur la figure 5.8d. Les courbes en trait gras représentent les cas crédibilistes et celles en trait fin les cas probabilistes (obtenus dans l'exemple 23).*

*On peut remarquer que les résultats sont assez similaires ce qui est normal puisque les données d'entrée des procédures* forward *et* backward *sont très proches. On peut cependant noter la petite différence sur l'intervalle* $[105, 130]$ *où les algorithmes crédibilistes ont tendance à favoriser l'action* chuter *(*$3^e$ *courbe des figures 5.4d et 5.8d) alors que les algorithmes probabilistes favorisent plutôt l'action* se relever *(*$4^e$ *courbe des figures 5.4d et 5.8d). Le cas crédibiliste est en accord avec la vérité terrain dans cet exemple.*

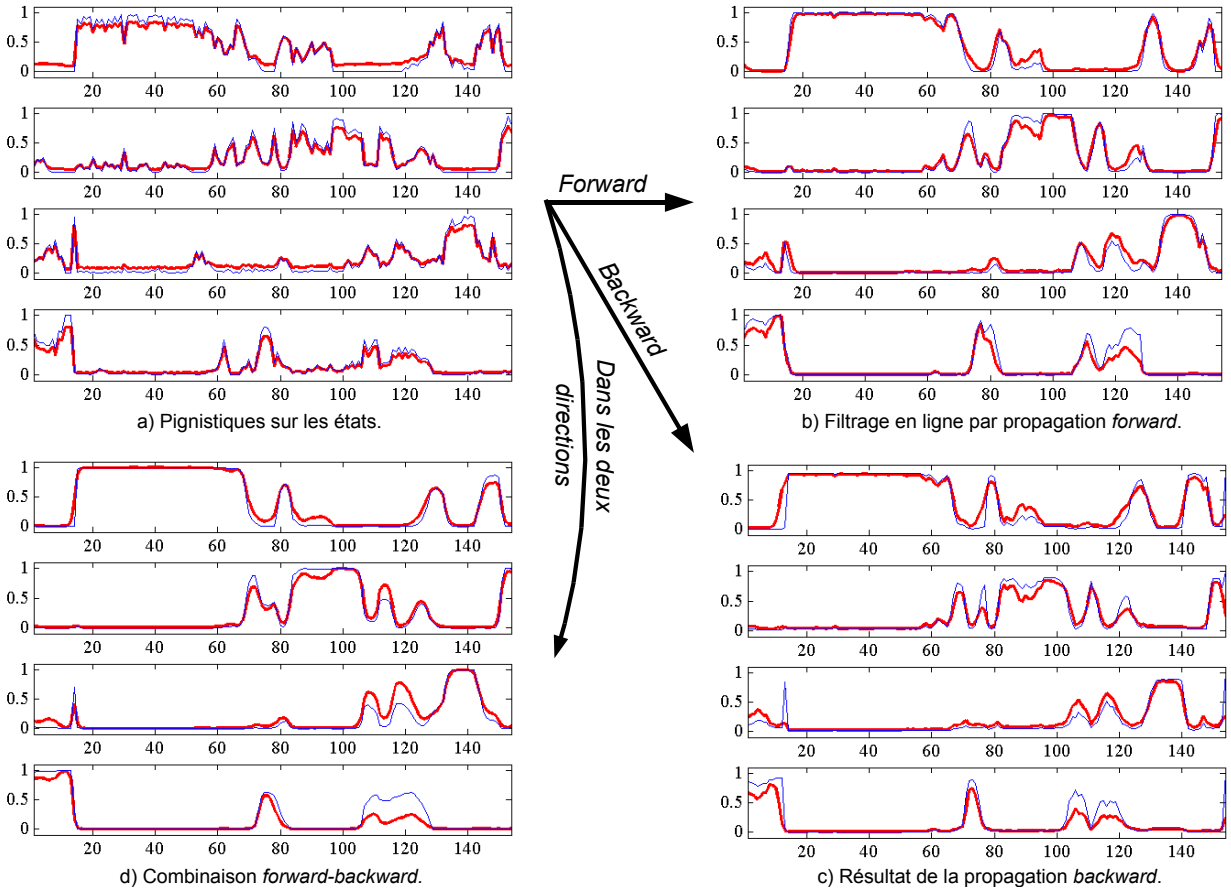



**FIG. 5.8 :** *Résultats du filtrage des croyances représentées en (a) : par les propagations* forward *(b) et* backward *(c) crédibilistes. La figure (d) donne le résultat de la combinaison conjonctive des deux propagations à chaque instant (propagation dans les deux directions). Un trait gras représente le cas crédibiliste et un trait fin représente le cas probabiliste (pour comparaison).*

### 5.3.4 Recherche de la meilleure séquence d'états

L'algorithme de Viterbi crédibiliste détermine la séquence d'états singletons « optimale » (au sens d'un critère prédéfini) $s^1, s^2 \dots s^T, s^t \in \Omega_t$ pouvant avoir générée la séquence des observations $\mathbf{O}_{1:T}$. La séquence optimale est déterminée parmi toutes les séquences possibles ($K^T$ cas pour $K$ états et $T$ observations) par des mécanismes de programmation dynamique comme dans le cas probabiliste.

Pour chaque état $s_j^t \in \Omega_t$, on doit déterminer le meilleur prédécesseur $s_i^{t-1} \in \Omega_{t-1}$. La recherche du prédécesseur fait intervenir à la fois les états à $t$ et $t-1$, c'est pourquoi on estime préalablement la BBA concernant les états sur l'espace produit $\Omega_{t-1} \times \Omega_t$. Cette BBA est obtenue en combinant les informations disponibles sur ces deux cadres de discernement à savoir (Fig. 5.9) :

- les fonctions de croyances conditionnelles $f_a^{\Omega_t}[S_i^{t-1}]$ décrivant les transitions,
- la fonction de croyances $f_b^{\Omega_t}$ relative aux observations,
- la fonction de croyances $f_\delta^{\Omega_{t-1}}$ sur les états à l'instant précédent (nous expliquerons par la suite comment nous obtenons cette croyance).

où $f \in \{pl, bel, m, q, b\}$.

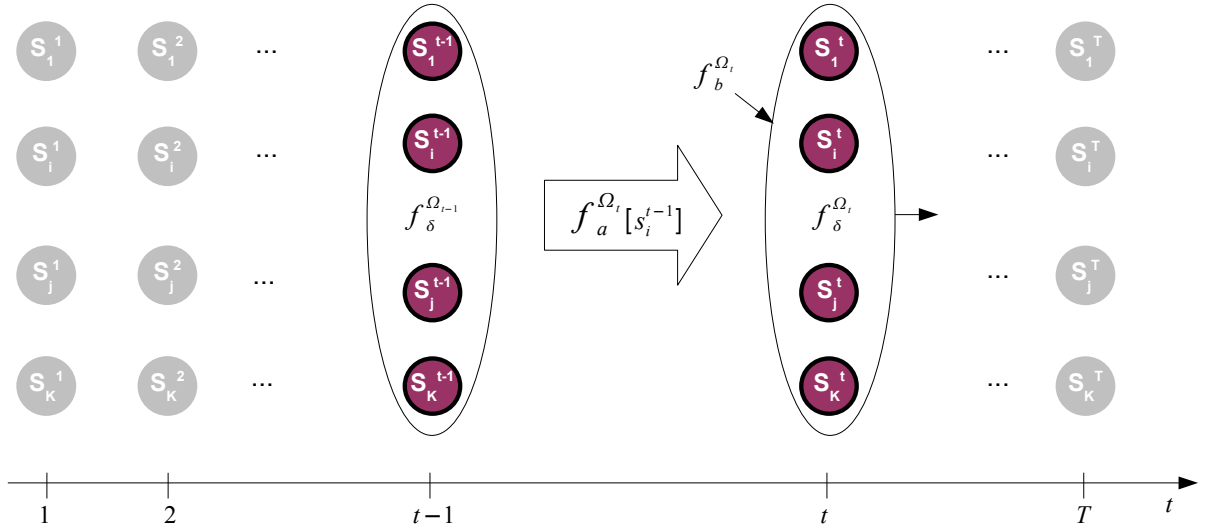



**FIG. 5.9 :** *Les informations disponibles à t : la distribution $f_{\alpha}^{\Omega_{t-1}}$ sur les états à $t-1$, les transitions $f_a^{\Omega_t}[s_i^{t-1}]$ et les croyances $f_b^{\Omega_t}$ relatives aux observations.*

#### 5.3.4.1 Synthèse de l'information courante

Les BBA $f_a^{\Omega_t}[S_i^{t-1}]$, $f_b^{\Omega_t}$ et $f_{\delta}^{\Omega_{t-1}}$ sont tout d'abord exprimées sur l'espace commun $\Omega_{t-1} \times \Omega_t$ (Fig. 5.9) :

– **Extension vide de $m_{\delta}^{\Omega_{t-1}}$ sur $\Omega_{t-1} \times \Omega_t$ :** la BBA de la variable $\delta$ à l'instant précédent $t-1$ est étendue avec l'opérateur d'extension vide (Eq. 2.37) sur l'espace produit $\Omega_{t-1} \times \Omega_t$ car la source de $m_{\delta}^{\Omega_{t-1}}$ ne connaît rien des éléments sur $\Omega_t$. On obtient :

$$m_{\delta}^{\Omega_{t-1}\uparrow\Omega_{t-1}\times\Omega_t}(C) = \begin{cases} m_{\delta}^{\Omega_{t-1}}(B) & \text{si } C = B \times \Omega_t \\ 0 & \text{sinon} \end{cases} \tag{5.48}$$

– **Extension vide de $q_b^{\Omega_t}$ sur $\Omega_{t-1} \times \Omega_t$ :** la communalité des observations à l'instant courant $t$ est étendue avec l'opérateur d'extension vide (Eq. 2.37) sur l'espace produit $\Omega_{t-1} \times \Omega_t$ car la source de $m_b^{\Omega_t}$ ne connaît rien des éléments sur $\Omega_{t-1}$. Pour cela, nous transformons d'abord la communalité en BBA (Eq. 2.7b) puis nous appliquons l'opérateur d'extension vide :

$$m_b^{\Omega_t\uparrow\Omega_t\times\Omega_{t-1}}(C) = \begin{cases} m_b^{\Omega_t}(B) & \text{si } C = B \times \Omega_{t-1} \\ 0 & \text{sinon} \end{cases} \tag{5.49}$$

La BBA obtenue est ensuite transformée en communalités (Eq. 2.7a).

– **Déconditionnement des transitions $q_a^{\Omega_t}[s_i^{t-1}]$ sur $\Omega_{t-1} \times \Omega_t$ :** les BBA définies sur $\Omega_t$ conditionnellement aux singletons $s_i^{t-1} \in \Omega_{t-1}$ sont déconditionnées sur l'espace produit $\Omega_t \times \Omega_{t-1}$ en utilisant l'opérateur de déconditionnement basé sur les communalités (Eq. 2.54). On obtient :

$$q_a^{\Omega_{t-1}\times\Omega_t}(S) \quad = \prod_{s_i^{t-1}\in\Omega_{t-1}} q_a^{\Omega_t}[s_i^{t-1}](v), \forall S \subseteq \Omega_t \times \Omega_{t-1} \tag{5.50}$$

et où $v = ((s_i^{t-1} \times \Omega_t) \cap S)^{\downarrow\Omega_t})$.

Les fonctions de croyances (BBA et communalités) peuvent maintenant être combinées par un opérateur conjonctif tel que la CRC (supposant les BBA distinctes) :

$$q_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}(S) = q_{\delta}^{\Omega_{t-1}\uparrow\Omega_{t-1}\times\Omega_t}(S) \cdot q_{a}^{\Omega_{t-1}\times\Omega_t}(S) \cdot q_{b}^{\Omega_t\uparrow\Omega_{t-1}\times\Omega_t}(S) \tag{5.51}$$

**Remarque 5.2.** *Dans l'expression 5.51, nous supposons les sources de BBA distinctes. Dans le cas contraire, il suffit d'appliquer l'opérateur prudent CCRC (§ 2.2.5, Eq. 2.64). Notons que si les BBA $q_a^{\Omega_t}[s_i^{t-1}]$ ne sont pas distinctes alors l'expression 5.50 n'est plus valable et il est alors nécessaire d'utiliser l'équation 2.45 pour déterminer toutes les BBA déconditionnées puis de combiner ces BBA en utilisant l'opérateur CCRC (Eq. 2.64) comme expliqué dans [Den07].*

#### 5.3.4.2 Recherche du prédécesseur

A présent que la connaissance sur $t-1$ et $t$ est connue, on peut déterminer le prédécesseur $\psi(s_j^t)' \in \Omega_{t-1}$ d'un état $s_j^t$. Le meilleur prédécesseur $\psi'(s_j^t)$ d'un état $s_j^t$ est celui qui maximise soit la *probabilité pignistique* soit la *plausibilité* (i.e. un critère de décision) de l'état $s_j^t$ après les étapes suivantes :

1. Conditionnement de la BBA[5] $m_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}$ par le prédécesseur « candidat » $s_i^{t-1}$,
2. Projection sur $\Omega_t$ de la BBA $m_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}$,
3. Calcul du critère de décision (probabilité pignistique et plausibilité) de chaque état sur $\Omega_t$.

On obtient donc :

$$\mathcal{P}_t^1[s_i^{t-1}](s_j^t) \;=\; \mathbf{BetP}^{\Omega_t}\{m_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}[s_i^{t-1}\times\Omega_t]^{\downarrow\Omega_t}\}(s_j^t) \tag{5.52}$$

ou encore, avec les plausibilités :

$$\mathcal{P}_t^2[s_i^{t-1}](s_j^t) \;=\; \mathbf{PlP}^{\Omega_t}\{m_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}[s_i^{t-1}\times\Omega_t]^{\downarrow\Omega_t}\}(s_j^t) \tag{5.53}$$

Le conditionnement par $s_i^{t-1}$ dans les équations 5.52-5.53 engendre généralement du conflit. Ce conflit représente la pertinence de l'hypothèse « *le prédécesseur de $s_j^t$ est $s_i^{t-1}$* » que l'on mémorise dans la variable $\Gamma_t(s_i^{t-1})$ à valeur dans $[0,1]$ :

$$\Gamma_t(s_i^{t-1}) = 1 - m_{\text{a}\delta\text{b}}^{\Omega_{t-1}\times\Omega_t}[s_i^{t-1}\times\Omega_t](\emptyset) \tag{5.54}$$

Plus le conflit est faible, plus l'hypothèse « *le prédécesseur de $s_j^t$ est $s_i^{t-1}$* » est pertinente et on exploitera cette information pour la prise de décision concernant le meilleur prédécesseur.

Pour chaque état $s_j^t$, on dispose donc d'un jeu de probabilités $\mathcal{P}_t^k[s_i^{t-1}](s_j^t), \forall s_i^{t-1} \in \Omega_{t-1}, k \in \{1,2\}$ ainsi qu'un ensemble de pertinences $\Gamma_t(s_i^{t-1}), \forall s_i^{t-1} \in \Omega_{t-1}$. Afin de déterminer, parmi tous les états $s_i^{t-1} \in \Omega_{t-1}$, le meilleur prédécesseur de l'état $s_j^t$, il suffit de chercher l'état $s_i^{t-1}$ maximisant la valeur $\mathcal{P}_t^k[s_i^{t-1}](s_j^t)$ pondérée par la pertinence associée à $s_i^{t-1}$ :

$$\psi_t'(s_j^t) = \underset{s_i^{t-1}\in\Omega_{t-1}}{\operatorname{argmax}} \left[\Gamma_t(s_i^{t-1}) \cdot \mathcal{P}_t^k[s_i^{t-1}](s_j^t)\right] \tag{5.55a}$$

$$\delta_t'(s_j^t) = \max_{s_i^{t-1}\in\Omega_{t-1}} \left[\Gamma_t(s_i^{t-1}) \cdot \mathcal{P}_t^k[s_i^{t-1}](s_j^t)\right] \tag{5.55b}$$

[5] Obtenue à partir de la communalité 5.51 en utilisant l'équation 2.7b.

où $\delta_t^{'}(s_j^t)$ et $\psi_t^{'}(s_j^t)$ mémorisent la valeur maximale de la transition du prédécesseur de $s_j^t$ vers $s_j^t$ ainsi que l'indice du prédécesseur.

Les équations 5.55a et 5.55b quantifient l'impact de l'hypothèse « *le prédécesseur de $s_j^t$ est $s_i^{t-1}$* ». Notons que l'ensemble des valeurs $\delta_t^{'}(s_j^t), \forall s_j^t \in \Omega_t$ ne forme pas une distribution.

#### 5.3.4.3 Propagation des résultats

On connaît le prédécesseur de chaque état $s_j^t$ à l'instant $t$. Il faut maintenant construire la BBA $m_\delta^{\Omega_t}$ qui sera utilisée à l'itération suivante $(t+1)$ et qui a pour rôle de propager les valeurs maximales $\delta_t^{'}(s_j^t)$ déterminées à $t$.

Pour construire la BBA $m_\delta^{\Omega_t}$, remarquons tout d'abord que plusieurs états $s_j^t$ peuvent avoir un prédécesseur commun ($\exists s_j^t, s_k^t \in \Omega_t / \psi_t^{'}(s_j^t) = \psi_t^{'}(s_k^t)$). De plus, on ne connaît pas quel état $s_j^t \in \Omega_t$ est le plus fiable à un instant $t$ (excepté éventuellement à l'instant $T$ où il faut choisir un état final).

Ainsi, nous proposons d'estimer la BBA $m_\delta^{\Omega_t}$ en projetant la BBA $m_\delta^{\Omega_{t-1}}$ sur $\Omega_t$ par l'algorithme *forward* mais en conditionnant la projection à l'ensemble des états prédecesseurs définis sur $t-1$ et noté $\mathcal{A}_{t-1}$ tel que[6] :

$$\mathcal{A}_{t-1} = \bigcup_{s_j^t \in \Omega_t} \psi^{'}(s_j^t) \tag{5.56}$$

Le résultat est ensuite combiné avec la BBA relative aux observations et la BBA $m_\delta^{\Omega_t}$ résultante est alors donnée par :

$$m_\delta^{\Omega_t} = m_b^{\Omega_t} \textcircled{\cap} \left( \left( m_\delta^{\Omega_{t-1} \uparrow \Omega_{t-1} \times \Omega_t} \textcircled{\cap} m_a^{\Omega_{t-1} \times \Omega_t} \right) [\Omega_t \times \mathcal{A}_{t-1}] \right)^{\downarrow \Omega_t} \tag{5.57}$$

De la sorte, si tous les prédécesseurs sont différents, le conditionnement est fait sur $\Omega_{t-1} \times \Omega_t$ et donc la projection n'a pas de parti pris alors que si des prédécesseurs sont identiques, la projection tient compte de cette situation et rend la BBA projetée plus spécifique.

L'expression 5.57 peut être simplifiée en utilisant l'équation de propagation *forward* basée sur les communalités et en conditionnant les propagations aux prédécesseurs :

$$\boldsymbol{q_\delta^{\Omega_t}(S_j^t) = \left( \sum_{S_i^{t-1} \subseteq \mathcal{A}_{t-1}} m_\delta^{\Omega_{t-1}}(S_i^{t-1}) \cdot q_a^{\Omega_t}[S_i^{t-1}](S_j^t) \right) \cdot q_b^{\Omega_t}(S_j^t), \quad \forall S_j^t \subseteq \Omega_t} \tag{5.58}$$

**Remarque 5.3.** *Le calcul de $m_\delta^{\Omega_t}$ ne prend pas directement en compte les valeurs $\delta_t^{'}(s_j^t)$ mais indirectement par le conditionnement. Nous proposons deux méthodes alternatives pour calculer $m_\delta^{\Omega_t}$ directement à partir des valeurs $\delta_t^{'}(s_j^t)$. Ces propositions dépendent de l'interprétation de $\delta_t^{'}(s_j^t)$ :*

- ***Alternative 1 : La valeur $\delta_t^{'}(s_j^t)$ représente la probabilité maximale de $s_j^t$.***
  *Dans ce cas, on doit normaliser les valeurs $\delta_t^{'}(s_j^t)$ telle que la somme sur $s_j^t \in \Omega_t$ soit unitaire permettant ainsi d'obtenir une distribution de probabilités. Ensuite, on peut transformer cette distribution en BBA consonante par application du Principe du Minimum d'Information (PMI, chapitre 2, paragraphe 2.2.4). La BBA résultante est alors assimilée à $m_\delta^{\Omega_t}$ qui diffèrera du résultat de l'équation 5.57.*

[6] La lettre $\mathcal{P}$ a été choisie pour noter les ensembles de probabilités aux équations 5.52 et 5.53. La lettre $\mathcal{A}$ signifie « ascendants » qui a ici le même sens que « prédécesseurs ».

*Si les valeurs $\delta_t^{'}(s_j^t)$ sont du même ordre de grandeur alors la BBA générée $m_\delta^{\Omega_t}$ sera proche de la BBA vide. Dans ce cas, l'a priori sur $\Omega_{t-1}$ utilisé dans l'équation de base (Eq. 5.51) sera minimum. Le même phénomène apparaît dans le cas probabiliste.*

– ***Alternative 2 : La valeur $\delta_t^{'}(s_j^t)$ représente la plausibilité maximale de $s_j^t$.***
*Dans ce cas, la BBA $m_\delta^{\Omega_t}$ est consonante. On en déduit alors sa distribution de plausibilités donnée par [DP82] :*

$$pl_\delta^{\Omega_t}(B) = \max_{s_j^t \in B} \ \delta_t^{'}(s_j^t) \tag{5.59}$$

*Ensuite, il suffit de transformer cette distribution de plausibilités (qui est aussi une distribution de possibilités) en BBA par application de l'équation 2.4b. La BBA résultante est alors assimilée à $m_\delta^{\Omega_t}$ qui diffèrera du résultat de l'équation 5.57.*
*Si les valeurs $\delta_t^{'}(s_j^t)$ sont du même ordre de grandeur alors la BBA générée $m_\delta^{\Omega_t}$ sera composée de deux éléments focaux, à savoir l'ensemble vide et le cadre tout entier tels que :*

$$\begin{aligned} m_\delta^{\Omega_t}(\emptyset) &= 1 - pl_\delta^{\Omega_t}(\Omega_t) \\ &= 1 - \max_{s_j^t \in \Omega_t} \ \delta_t^{'}(s_j^t) \end{aligned} \tag{5.60}$$

*Cette alternative peut donc s'avérer problématique lorsque les valeurs $\delta_t^{'}(s_j^t)$ sont faibles car dans ce cas un conflit élevé est réinjecté dans l'équation 5.51 utilisée à l'itération suivante.*

#### 5.3.4.4 Détermination de la séquence d'états optimale et de son adéquation avec un modèle

Nous cherchons à recouvrir la séquence d'états optimale, au sens du maximum de probabilité pignistique ou de plausibilité, à partir de l'ensemble des valeurs $\psi_t^{'}$ et à quantifier son adéquation avec le modèle de CrHMM.

**Séquence d'états optimale** Elle est notée $s_*^1, s_*^2, \ldots s_*^t, \ldots s_*^T$. A la dernière observation (à l'instant $T$), le meilleur état $s_*^T$ est donné par :

$$s_*^T = \underset{s_j^T \in \Omega_T}{\operatorname{argmax}} \ \mathbf{BetP}^{\Omega_T}\{m_\delta^{\Omega_T}\}(s_j^T) \tag{5.61}$$

A partir de cet état, le meilleur prédécesseur $s_*^{T-1}$ est choisi en utilisant la valeur $\psi_T^{'}(s_*^T)$ (Eq. 5.55a). La valeur $\psi_{T-1}^{'}(s_*^{T-1})$ correspond quant à elle au meilleur état à $T-2$ lorsque $s_*^{T-1}$ est sélectionné à $T-1\ldots$ Ce processus, appelé *rétro-propagation*, est réitéré jusqu'à $t=1$ permettant de recouvrir la séquence d'états optimale :

$$\boldsymbol{s_*^t = \psi_{t+1}^{'}(s_{t+1}^*)} \tag{5.62}$$

Ce processus est similaire à celui utilisé dans le cas des HMM probabilistes.

**Adéquation de la séquence d'états optimale avec un modèle de CrHMM** Dans le cadre de la classification de séquences, on cherche à quantifier l'adéquation de la séquence optimale avec le modèle de la séquence utilisé. Pour cela, on propose de calculer la plausibilité $\mathcal{Q}_T(s_*^T, \lambda)$ de la séquence optimale conditionnellement à un modèle de CrHMM et se terminant en $s_*^T$ :

$$\mathcal{Q}_T(s_*^T, \lambda) = pl^{\Omega_{1:T}}[\lambda](\{(s_*^1, s_*^2, \ldots s_*^T)\}) \tag{5.63}$$

où $(s_*^1, s_*^2, \dots s_*^T)$ est un $T$-uplet et $\Omega_{1:T} = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_T$ et le produit cartésien des cadres de discernement où sont définis les éléments du $T$-uplet.

Pour déterminer l'expression analytique de ce critère, on suppose que les plausibilités $pl^{\Omega_t}[\lambda](s_*^t), \forall t$ sont indépendantes[7]. Rappelons par ailleurs que la plausibilité $pl_\delta^{\Omega_t}$ d'un état $s_*^t$ faisant partie de la séquence optimale est calculée à partir de la BBA $m_\delta^{\Omega_t}$ en utilisant l'équation 2.4a (on omet volontairement le conditionnement par $\lambda$). La plausibilité de la séquence optimale se terminant en $s_*^T$ est alors donnée par :

$$\mathcal{Q}_T(s_*^T, \lambda) = \prod_{t=1}^{T} pl_\delta^{\Omega_t}(s_*^t) \tag{5.64}$$

Par ailleurs, on peut remarquer qu'il n'est pas forcément nécessaire de procéder à la rétro-propagation pour quantifier la plausibilité de la séquence optimale. En effet, à chaque instant, on a :

$$\mathcal{Q}_t(s_j^t, \lambda) = \mathcal{Q}_{t-1}(\psi_t^{'}(s_j^t), \lambda) \cdot pl_\delta^{\Omega_t}(s_j^t) \tag{5.65}$$

et lorsque la dernière observation est atteinte à $t = T$, le meilleur état $s_*^T$ peut donc être choisi en maximisant le critère $\mathcal{Q}_T(., \lambda)$ :

$$\boldsymbol{s_*^T = \underset{s_j^T \in \Omega_T}{\operatorname{argmax}}\ \mathcal{Q}_T(s_j^T, \lambda)} \tag{5.66}$$

La valeur $\mathcal{Q}_T(s_*^T, \lambda)$ associée quantifie ainsi directement la plausibilité de la séquence (Eq. 5.63).

Notons que pour parer aux limitations des machines actuelles lors des multiplications successives dans l'équation 5.64, nous pouvons calculer le logarithme du critère et ainsi obtenir la « log-plausibilité » de la séquence optimale :

$$\boldsymbol{\mathcal{Q}_t(s_j^t, \lambda) = \mathcal{Q}_{t-1}(\psi_t^{'}(s_j^t), \lambda) + \log(pl_\delta^{\Omega_t}(s_j^t))} \tag{5.67}$$

et à l'initialisation, on a $\mathcal{Q}_1(s_j^1, \lambda) = 0, \forall s_j^t \in \Omega_1$. Remarquons pour finir qu'au lieu des plausibilités, nous pourrions utiliser la probabilité pignistique $\mathbf{BetP}^{\Omega_t}\{pl_\delta^{\Omega_t}\}$ (les mêmes équations pourraient être utilisées).

**Exemple 26.** *On reprend l'exemple 25 concernant la reconnaissance d'une séquence de saut en hauteur composée de* $4$ *états :* courir *(*$s_1$*),* sauter *(*$s_2$*),* chuter *(*$s_3$*) et* se relever *(*$s_4$*), et on note* $\Omega_t = \{s_1^t, s_2^t, s_3^t, s_4^t\}$.

*La figure 5.10a présente le résultat de l'algorithme de Viterbi crédibiliste pour la reconnaissance de la séquence d'états à partir des BBA observées de la figure 5.8a. La figure 5.10b illustre le résultat fourni par la maximisation a posteriori (Eq. 5.12) sur les probabilités pignistiques de* $q_\gamma^{\Omega_t}$ *données à la figure 5.8d. La comparaison qualitative entre les figures 5.10a-b permet de mettre en évidence l'intérêt du Viterbi crédibiliste par rapport à une simple maximisation instant par instant sans prise en compte des prédécesseurs. Les résultats quantitatifs, résumés dans le tableau 5.5 (avec les indices de rappel (*$R$*), précision (*$P$*) et mesure* $F_1$ *de la détection des actions), vont aussi dans ce sens.*

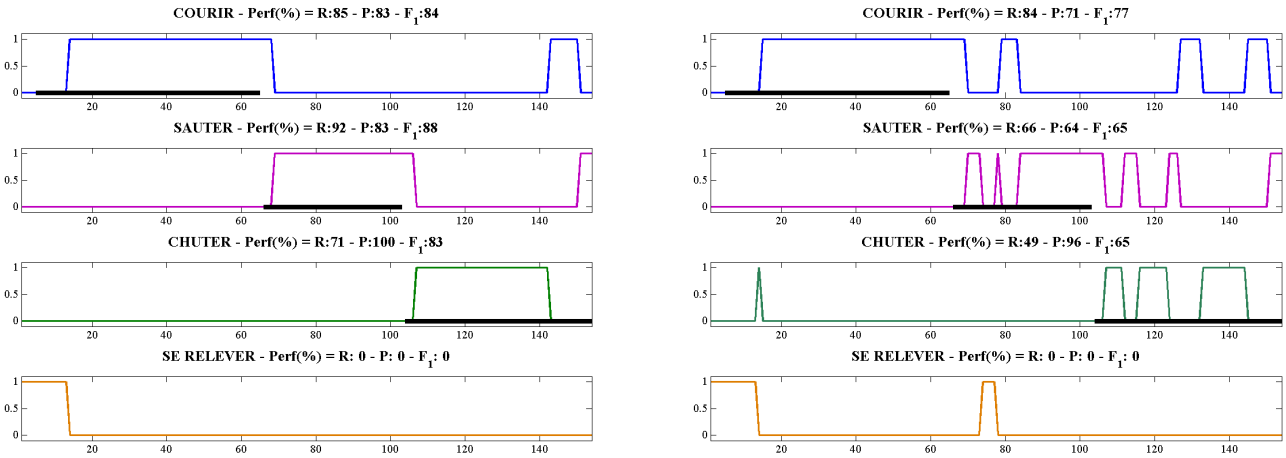
*La différence des résultats entre les algorithmes de Viterbi crédibiliste et probabiliste (Fig. 5.6) peut être appréciée qualitativement sur la figure 5.11 et quantitativement dans le tableau 5.5.*

*La figure 5.12 illustre l'évolution de la log-plausibilité (Fig. 5.12a) et de la log-vraisemblance (Fig. 5.12b) de la séquence d'états optimale déterminée respectivement par les algorithmes de Viterbi*

[7]Plus précisément on considère l'indépendance *faible* (*weak independance*) [Sha76] qui permet d'écrire $pl^{\Omega_t \times \Omega_{t-1}}(\{(B, C)\}) = pl^{\Omega_t}(B) \cdot pl^{\Omega_{t-1}}(C)$.

*crédibiliste et probabiliste. On peut remarquer une tendance similaire dans l'évolution cependant, le critère crédibiliste met en évidence des paliers lorsque l'état est plutôt stable ce qui pourrait s'avérer intéressant pour la détection de changement.*

*La figure 5.13 représente la probabilité pignistique* $\mathbf{BetP}^{\Omega_t}\{m_\delta^{\Omega_t}\}(s_j^t)$ *sur les états déterminées à partir des BBA* $m_\delta^{\Omega_t}$ *propagées dans l'algorithme de Viterbi crédibiliste (courbe de couleur rouge et en gras). Les probabilités* $\delta$ *dans le cas probabiliste sont aussi illustrées (courbe de couleur bleue en trait fin) permettant d'expliquer la différence des séquences d'états optimales (Fig. 5.11) et notamment sur l'intervalle* $[105, 135]$.




a) Décodage de Viterbi crédibiliste.

b) Maximisation a posteriori sur la combinaison *forward-backward crédibiliste*.

**FIG. 5.10 :** *Résultats de l'algorithme de Viterbi crédibiliste (a) pour la reconnaissance de la séquence d'états la plus probable à partir des observations de la figure 5.8a. Comparaison avec la détermination du meilleur état (b) en utilisant la maximisation a posteriori (Eq. 5.12) sur les probabilités pignistiques de* $q_\gamma^{\Omega_t}$ *données à la figure 5.8d.*

| | course | | | saut | | | chute | | |
|---|---|---|---|---|---|---|---|---|---|
| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
| HMM | 84 | 80 | 82 | 82 | 82 | 82 | 31 | 94 | 47 |
| $\gamma$ proba. | 84 | 72 | 77 | 63 | 75 | 69 | 24 | 92 | 38 |
| CrHMM | **85** | **83** | **84** | **92** | **83** | **88** | **71** | **100** | **83** |
| $q_\gamma$ créd. | 84 | 71 | 77 | 66 | 64 | 65 | 49 | 96 | 65 |

**TAB. 5.5 :** *Indices de performance : rappel (R), précision (P) et mesure* $F_1$ *en* $\%$ *sur la détection des actions par le décodage de Viterbi probabiliste (ligne 1), par maximisation de la probabilité* $\gamma_t$ *a posteriori (ligne 2), par le décodage de Viterbi crédibiliste (ligne 3) et par maximisation de la probabilité pignistique de* $q_\gamma^{\Omega_t}$ *a posteriori.*

### 5.3.5 Classification de séquence d'observations

Disposant de $N$ modèles de CrHMM $\lambda_1, \lambda_2, \ldots, \lambda_n, \ldots, \lambda_N$ représentant chacun un type de séquence particulier, on cherche à connaître le modèle $\lambda_*$ correspondant le mieux aux observations

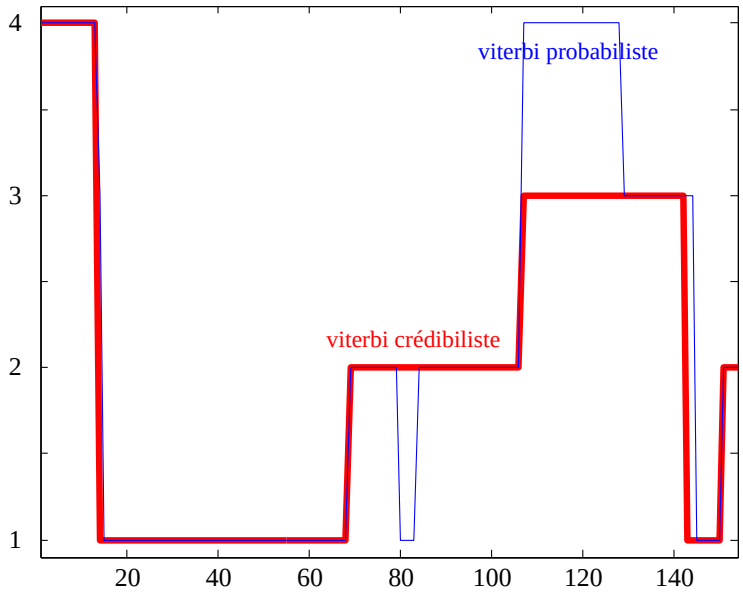
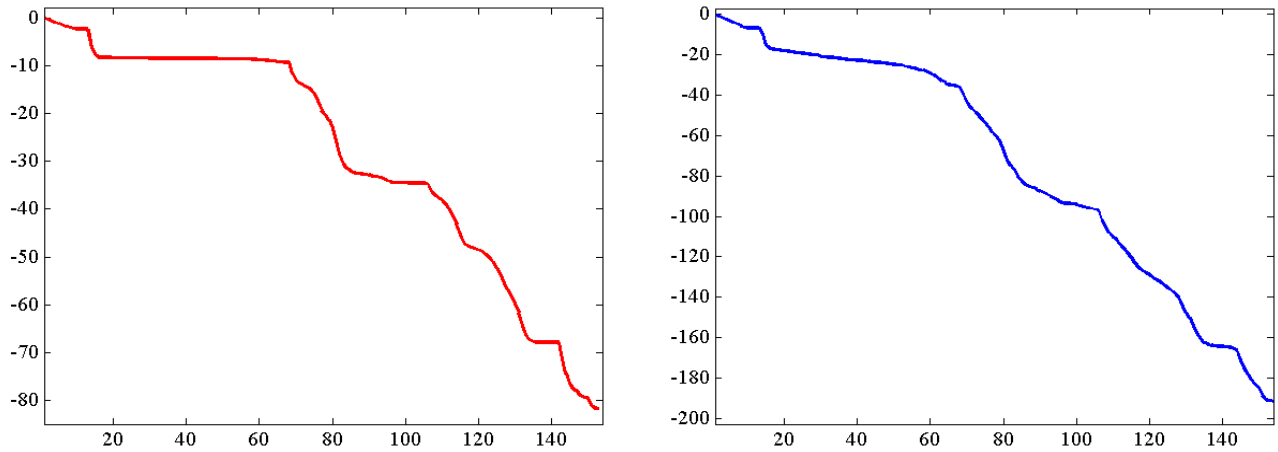



**FIG. 5.11 :** *Superposition des décodages de Viterbi crédibiliste et probabiliste pour la reconnaissance du saut en hauteur (Figs. 5.8-5.10) sur une vidéo composée de* 154 *images. Le saut en hauteur est décrit par une séquence de quatre actions à savoir* courir *(valeur* 1 *sur l'axe des ordonnées de la figure),* sauter *(valeur* 2*),* chuter *(valeur* 3*) et* se relever *(valeur* 4*).*




a) Valeur du critère d'inférence lors du décodage de Viterbi **crédibiliste**.

b) Valeur du critère d'inférence lors du décodage de Viterbi **probabiliste**.

**FIG. 5.12 :** *Evolution des critères d'inférence pour le cas crédibiliste (a) et probabiliste (b) pour la reconnaissance du saut en hauteur (Figs. 5.8-5.10).*

mesurées $\mathbf{O}_{1:T}$, i.e. on cherche à classer une séquence d'observations. Chaque modèle de CrHMM $\lambda$ est caractérisé par un jeu de BBA conditionnelles, d'un modèle d'observations et d'un a priori éventuel.

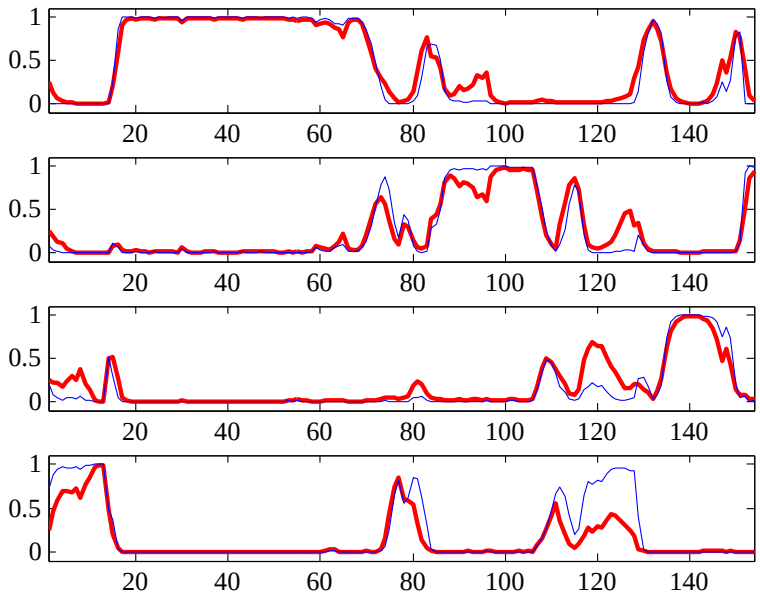



**FIG. 5.13 :** *Evolution des probabilités pignistiques* $\mathbf{BetP}^{\Omega_t}\{m_\delta^{\Omega_t}\}$ *calculées à partir des BBA* $m_\delta^{\Omega_t}$ *propagées dans l'algorithme de Viterbi crédibiliste (courbe de couleur rouge et en gras). Les probabilités* $\delta$ *dans le cas probabiliste sont aussi illustrées (courbe de couleur bleue en trait fin).*

#### 5.3.5.1 Interprétation du conflit

On considère le cas d'une classification « en ligne » comme dans le cadre probabiliste. Le critère de classification repose sur la propagation *forward* (Eq. 5.39) faisant intervenir un processus de prédiction et un de mise à jour (Eq. 5.41) comme nous le rappelons ci-dessous :

$$\begin{aligned} q_\alpha^{\Omega_t}(S_j^t) &= \hat{q}_\alpha^{\Omega_t}(S_j^t) \cdot q_b^{\Omega_t}(S_j^t) \\ &= \Big( \sum_{S_i^{t-1} \subseteq \Omega_{t-1}} m_\alpha^{\Omega_{t-1}}(S_i^{t-1}) \cdot q_a^{\Omega_t}[S_i^{t-1}](S_j^t) \Big) \cdot q_b^{\Omega_t}(S_j^t), \quad \forall S_j^t \subseteq \Omega_t \end{aligned}$$

Suite à la propagation *forward*, la masse de conflit est égale à :

$$m_\alpha^{\Omega_t}(\emptyset) = 1 + \sum_{S_j^t \subseteq \Omega_t, S_j^t \neq \emptyset} (-1)^{|S_j^t|} \cdot \hat{q}_\alpha^{\Omega_t}(S_j^t) \cdot q_b^{\Omega_t}(S_j^t) \tag{5.68}$$

et notons que cette quantité correspond à $(\hat{m}_\alpha^{\Omega_t} \textcircled{\cap} m_b^{\Omega_t})(\emptyset)$.

Cette masse de conflit fait intervenir à la fois le conflit apporté directement par la prédiction et le conflit entre la prédiction et les observations. Le premier type de conflit est supposé nul car nous avons discuté de sa redistribution à la fin du paragraphe 5.3.2.1. Le deuxième type de conflit est intéressant car il quantifie l'*inadéquation du modèle de CrHMM* $\lambda$ *utilisé avec les observations* $\mathbf{O}_{1:T}$. Plus ce conflit est élevé, moins le modèle de CrHMM correspond aux observations ce qui nous permet d'envisager un critère de classification basé sur cette information de conflit.

**Remarque 5.4.** *Si le conflit apporté directement par la prédiction est non nul, nous devons* ne pas *en tenir compte dans l'équation 5.68 afin de ne garder que le conflit pertinent qui apparaît entre la*

*prédiction et les observations. Pour éliminer la part de conflit du premier type, il suffit de normaliser la communalité $\hat{q}_\alpha^{\Omega_t}$ utilisée dans l'équation 5.68. On obtient alors :*

$$m_\alpha^{\Omega_t}(\emptyset) = 1 + \frac{1}{K_\alpha^t} \cdot \sum_{S_j^t \subseteq \Omega_t, S_j^t \neq \emptyset} (-1)^{|S_j^t|} \cdot \hat{q}_\alpha^{\Omega_t}(S_j^t) \cdot q_b^{\Omega_t}(S_j^t) \tag{5.69}$$

*où le coefficient $K_\alpha^t$ est donné par :*

$$\begin{aligned} K_\alpha^t &= 1 - \left(1 + \sum_{C \subseteq \Omega_t, C \neq \emptyset} (-1)^{|C|} \cdot \hat{q}_\alpha^{\Omega_t}(C)\right) \\ &= - \sum_{C \subseteq \Omega_t, C \neq \emptyset} (-1)^{|C|} \cdot \hat{q}_\alpha^{\Omega_t}(C) \end{aligned} \tag{5.70}$$

#### 5.3.5.2 Un critère de classification crédibiliste basé sur le conflit

Comme nous l'avons vu, plus le conflit entre la prédiction et les observations est élevé, moins le modèle de CrHMM correspond aux observations. A partir de cette interprétation, nous générons une *mesure globale de conflit* $\mathcal{L}_c(\lambda)$ entre un modèle de CrHMM $\lambda$ et des observations $\mathbf{O}_{1:T}$ définie par :

$$\boldsymbol{\mathcal{L}_c(\lambda) = \frac{1}{T} \sum_{t=1}^{T} \log(m_\alpha^{\Omega_t}[\lambda](\emptyset))} \tag{5.71}$$

où la quantité $m_\alpha^{\Omega_t}[\lambda](\emptyset)$ est donnée par l'équation 5.68 (ou par l'équation 5.69 si la prédiction présente du conflit). Le modèle $\lambda_*$ correspondant le mieux aux observations mesurées est alors celui qui *minimise* le conflit sur la séquence des observations :

$$\boldsymbol{\lambda^* = \underset{n}{\text{argmin}} \ \mathcal{L}_c(\lambda_n)} \tag{5.72}$$

Le point intéressant à noter est que ce critère est similaire, sur le principe, à celui utilisé dans le cadre probabiliste (Eq. 5.29). Pour comparer ces deux critères, rappelons que la normalisation de la BBA $m_\alpha^{\Omega_{t-1}}[\lambda]$ consiste à diviser la valeur de chaque élément focal par l'opposé du conflit (Eq. 2.3) :

$$\begin{aligned} m_{\alpha,*}^{\Omega_{t-1}}[\lambda](B) &= \frac{m_\alpha^{\Omega_{t-1}}[\lambda](B)}{\mathbf{1 - m_\alpha^{\Omega_{t-1}}[\lambda](\emptyset)}} \\ &= \frac{m_\alpha^{\Omega_{t-1}}[\lambda](B)}{\sum_{\substack{C \subseteq \Omega_{t-1} \\ C \neq \emptyset}} m_\alpha^{\Omega_{t-1}}[\lambda](C)} \end{aligned} \tag{5.73}$$

Or, dans la version probabiliste, nous avons vu que la normalisation de la variable *forward* $\alpha_{t-1}$ (Eq. 5.22-5.23) est donnée par :

$$\alpha_{t-1}^*(s_i^{t-1}) = \frac{\alpha_{t-1}(s_i^{t-1})}{\sum_{\mathbf{k=1}}^{\mathbf{N}} \alpha_\tau(\mathbf{s_k^\tau})}$$

Par conséquent, le coefficient de normalisation de la distribution de probabilités $\alpha_t$ pourrait être interprété comme une « probabilité » sur l'ensemble vide. Or, cette probabilité est supposée ne pas exister. Dans le cadre crédibiliste, la masse sur l'ensemble vide $m_\alpha^{\Omega_t}[\lambda](\emptyset)$ a un sens, à savoir le *conflit entre le modèle de CrHMM $\lambda$ et les observations*.

#### 5.3.5.3 Distance entre modèles de CrHMM

Dans certaines applications, de nombreux modèles de CrHMM peuvent coexister. Cependant, l'inférence, c'est à dire le mécanisme permettant de sélectionner le meilleur modèle, est d'autant plus long et difficile que le nombre de modèles est important.

Dans [JR85, Rab89, VHS+02], un critère de distance entre deux modèles de HMM probabilistes est décrit. Si la distance est faible alors cela signifie que les deux modèles sont proches et que l'un des deux peut être supprimé. Nous pouvons de la même manière définir un critère de distance dans le cadre des CrHMM.

Soient deux modèles de CrHMM notés $\lambda_1$, $\lambda_2$ et un lot d'observations $\mathbf{O}_{1:T}$ supposées avoir été générées par le modèle $\lambda_2$. Nous pouvons utiliser le critère $\mathcal{L}_c(\lambda)$ proposé précédemment pour calculer la distance $d_1(\lambda_1, \lambda_2)$ entre les deux modèles :

$$d_1(\lambda_1, \lambda_2) = \frac{1}{T}(\mathcal{L}_{c,2}(\lambda_1) - \mathcal{L}_{c,2}(\lambda_2)) \tag{5.74}$$

Cette mesure de distance quantifie l'adéquation du modèle $\lambda_1$ aux données générées par $\lambda_2$ relativement à l'adéquation du modèle $\lambda_2$ vis à vis de ses données (le fait que la source des observations soit $\lambda_2$ est indiqué par l'indice « 2 » sur le critère $\mathcal{L}_{c,2}$). Remarquons que cette mesure peut être interprétée comme une approximation de la divergence de Kullback-Leibler [JR85] utilisée en théorie de l'information [Sha48].

Le problème avec la mesure précédente est qu'elle n'est pas symétrique. Une version symétrisée est donnée par :

$$d_2(\lambda_1, \lambda_2) = \frac{d_1(\lambda_1, \lambda_2) + d_1(\lambda_2, \lambda_1)}{2} \tag{5.75}$$

Nous pouvons imaginer utiliser cette mesure pour la classification de séquences basée sur les similarités crédibilistes de la même manière que dans [BMF04, DM04].

### 5.3.6 Vers l'apprentissage des paramètres

Dans les sections précédentes, les paramètres des CrHMM étaient supposés connus. Comme dans le cadre des HMM probabilistes, il serait intéressant de définir une méthode d'estimation de ces paramètres par apprentissage. Cependant, la complexité est plus importante dans le cas crédibiliste car les fonctions de croyance n'ont pas les mêmes propriétés mathématiques que les probabilités et de plus, les croyances sont définies sur des espaces de grande taille.

Le problème de l'apprentissage concerne trois ensembles de paramètres :

1. **La matrice de transition crédibiliste ($q_a^{\Omega_t}[S_i^{t-1}]$)**

   Nous proposons de définir un équivalent crédibiliste à la variable $\xi_t$ utilisée en probabilité (Eq. 5.35). Puis, cette nouvelle variable est moyennée sur les séquences d'observations formant l'ensemble d'apprentissage.

2. **L'a priori sur les états à $t = 1$ ($q_\pi^{\Omega_1}$)**

   Nous supposons que l'a priori sur les états est généralement vide. Cependant, une méthode basée sur la variable $q_a^{\Omega_t}[S_i^{t-1}]$ est proposée pour estimer cette distribution.

3. **Les modèles d'observation liant les observations $\mathrm{O}_t$ aux croyances sur les états**

   Nous proposons des pistes pour l'estimation des paramètres des mélanges Gaussiens. Ces propositions constituent un travail préliminaire pour une future étude.

#### 5.3.6.1 Estimation de la matrice de transition

La matrice de transition crédibiliste, de taille $K \times 2^K$ (avec $K$ le nombre d'états), contient les BBA $q_a^{\Omega_t}[s_i^{t-1}], \forall s_i^{t-1} \in \Omega_{t-1}$ qui représentent les croyances des sous-ensembles d'états $S_j^t \subseteq \Omega_t$ conditionnellement aux états $s_i^{t-1} \subseteq \Omega_{t-1}$. Les croyances conditionnelles aux sous-ensembles d'états sont obtenues par un opérateur disjonctif à partir des croyances conditionnelles aux singletons.

De la même manière que dans le cas probabiliste, nous cherchons à estimer la matrice de transition crédibiliste *moyenne* vis à vis de données d'apprentissage constituées d'observations qui correspondent au modèle de CrHMM à estimer. Pour cela, nous calculons tout d'abord la BBA sur $\Omega_{t-1} \times \Omega_t$ prenant en compte les informations disponibles à deux instants consécutifs $t$ et $t-1$.

**Combinaison des informations sur deux instants consécutifs** Dans le cas d'une estimation « en ligne » des transitions, il suffit de combiner les informations disponibles aux instants $t$ et $t-1$ :

- les transitions $q_a^{\Omega_t}[S_i^{t-1}]$,
- les communalités conditionnelles aux observations $q_b^{\Omega_t}$ en $t$,
- les communalités à l'instant précédent $q_\alpha^{\Omega_{t-1}}$.

Pour cela, nous adoptons la même méthodologie que pour l'équation 5.51 ce qui mène à l'expression suivante :

$$q_{\xi_{\text{on}}}^{\Omega_{t-1}\times\Omega_t}(S) = q_\delta^{\Omega_{t-1}\uparrow\Omega_{t-1}\times\Omega_t}(S) \cdot q_a^{\Omega_{t-1}\times\Omega_t}(S) \cdot q_b^{\Omega_t\uparrow\Omega_{t-1}\times\Omega_t}(S) \tag{5.76}$$

définie $\forall S \subseteq \Omega_{t-1} \times \Omega_t$. Nous pouvons remarquer que le calcul de la marginale de $q_{\xi_{\text{on}}}^{\Omega_{t-1}\times\Omega_t}$ sur $\Omega_t$ (avec l'équation 2.41) redonne la variable *forward* comme dans le cas probabiliste :

$$q_\alpha^{\Omega_t} = q_{\xi_{\text{on}}}^{\Omega_{t-1}\times\Omega_t\downarrow\Omega_t} \tag{5.77}$$

Comme on peut le voir sur l'équation 5.76, l'estimation de la matrice de transition, modélisée par $q_a^{\Omega_t}$, fait appel à cette même matrice et par conséquent, l'estimation doit être intégrée dans un processus itératif comme dans le cas probabiliste [Rab89].

Dans le cas d'une estimation « hors ligne », on tient compte des communalités futures $q_\beta^{\Omega_{t+1}}$ et on obtient une variable $q_{\xi_{\text{off}}}^{\Omega_t\times\Omega_{t+1}}$ (version ***off**-line*) obtenue en combinant la version en ligne $q_{\xi_{\text{on}}}^{\Omega_t\times\Omega_{t+1}}$ (décalée d'une itération) avec la connaissance de la variable *backward* $q_\beta^{\Omega_{t+1}}$ :

$$q_{\xi_{\text{off}}}^{\Omega_t\times\Omega_{t+1}}(S) = q_{\xi_{\text{on}}}^{\Omega_t\times\Omega_{t+1}}(S) \cdot q_\beta^{\Omega_{t+1}\uparrow\Omega_t\times\Omega_{t+1}}(S) \tag{5.78a}$$

$$= q_\alpha^{\Omega_t\uparrow\Omega_t\times\Omega_{t+1}}(S) \cdot q_a^{\Omega_t\times\Omega_{t+1}}(S) \cdot q_b^{\Omega_{t+1}\uparrow\Omega_t\times\Omega_{t+1}}(S) \cdot q_\beta^{\Omega_{t+1}\uparrow\Omega_t\times\Omega_{t+1}}(S) \tag{5.78b}$$

$\forall S \subseteq \Omega_t \times \Omega_{t+1}$. Notons que la marginale de $q_{\xi_{\text{off}}}^{\Omega_t\times\Omega_{t+1}}$ sur $\Omega_t$ redonne la variable crédibiliste $q_\gamma^{\Omega_t}$ :

$$q_\gamma^{\Omega_t} = q_{\xi_{\text{off}}}^{\Omega_t\times\Omega_{t+1}\downarrow\Omega_t} \tag{5.79}$$

Remarquons par ailleurs que si $q_\beta^{\Omega_{t+1}}(S) = 1, \forall S \subseteq \Omega_{t+1}$ (la variable *backward* est égale à la fonction vide) alors on retrouve la version en ligne $q_{\xi_{\text{on}}}^{\Omega_t\times\Omega_{t+1}}$.

**Une estimation des transitions crédibilistes** Pour une séquence d'observations $\mathbf{O}_{1:T}$ donnée et reflétant l'état d'un système à analyser, il existe une distribution de communalités $q_{\xi_{\text{off}}}^{\Omega_t\times\Omega_{t+1}}$ à chaque instant $t$. Nous pouvons utiliser un opérateur de consensus comme la *moyenne* afin d'estimer la matrice

de transition crédibiliste $\overline{q}_a^{\Omega_t \times \Omega_{t+1}}$ sur l'ensemble des observations. Dans le cas d'un apprentissage hors ligne, on obtient :

$$\overline{q}_a^{\Omega_{t+1}}[s_i^t] \quad = \quad \left( \frac{1}{T} \sum_{t=1}^{T} q_{\xi_{\text{off}}}^{\Omega_t \times \Omega_{t+1}} \right) [s_i^t]^{\downarrow \Omega_t} \tag{5.80}$$

où une moyenne de communalités est une communalité. Cette méthode n'est pas sans rappeler celle qui a été proposée dans le cadre de l'estimation des paramètres du filtre temporel crédibiliste (Eq. 3.26).

Un problème avec cette méthode est l'effet de spécialisation. En effet, la variable $q_{\xi_{\text{off}}}^{\Omega_t \times \Omega_{t+1}}$ est composée de quatre conjonctions et par conséquent, la valeur de la croyance sur les éléments focaux de faible cardinalité sera généralement élevée alors que celle sur les éléments focaux de plus grande cardinalité sera généralement faible. De plus, il se peut que certaines transitions n'apparaissent pas ou très peu fréquemment dans l'ensemble des observations utilisées pour l'estimation. Comme dans le cas probabiliste, une solution consiste à imposer une croyance minimale sur chaque transition.

#### 5.3.6.2 Estimation de l'a priori crédibiliste

Les fonctions de croyance permettent de choisir un a priori vide $\overline{q}_\pi^{\Omega_1}$ modélisant le manque d'information sur les états $s_j^t \in \Omega_1$ à l'instant initial ($t = 1$). Il est par ailleurs possible d'estimer statistiquement l'a priori crédibiliste sur un sous-ensemble $S_i^t$ en utilisant les résultats précédents.

Nous avons fait remarquer (Eq. 5.77) que le calcul de la marginale de $q_{\xi_{\text{on}}}^{\Omega_{t-1} \times \Omega_t}$ sur $\Omega_t$ redonne la variable *forward* $q_\alpha^{\Omega_t}$. Nous pouvons exploiter ce résultat pour l'estimation de l'a priori en projetant sur $\Omega_1$ la communalité moyenne $\overline{q}_a^{\Omega_1 \times \Omega_2}$ calculée à l'équation 5.80 :

$$\overline{q}_\pi^{\Omega_1} = \overline{q}_a^{\Omega_1 \times \Omega_2 \downarrow \Omega_1} \tag{5.81}$$

et cette expression impose de disposer de plusieurs séquences d'observations (afin de calculer une moyenne). L'estimation de l'a priori par cette expression fait intervenir la variable *forward* en $t = 1$, i.e. $q_\alpha^{\Omega_1}$, or, celle-ci fait justement appel à l'a priori recherché. Par conséquent, il sera nécessaire d'inclure cette expression dans un algorithme d'estimation itératif comme dans le cas probabiliste [Rab89].

#### 5.3.6.3 Vers l'estimation des paramètres des modèles Gaussiens des observations

Dans le cas probabiliste, un mélange Gaussien des observations est associé à chaque état (Eq. 5.30). Le problème dans le cas crédibiliste est que, à chaque instant, nous devons lier les observations numériques non seulement aux états singletons $s_j^t \in \Omega_t$ mais aussi aux sous-ensembles d'états $S_j^t \subseteq \Omega_t$. Deux solutions sont possibles : exploiter les informations sur chacun des singletons pour en déduire les croyances sur les sous-ensembles ou bien, modéliser tous les singletons et sous-ensembles d'états par un mélange Gaussien.

**Exploiter les méthodes statistiques probabilistes** Dans la majorité des tests réalisés dans cette thèse avec le CrHMM, nous avons utilisé des mélanges Gaussiens probabilistes pour lesquels des méthodes d'estimation robustes ont été éprouvées depuis longtemps. Ces mélanges fournissent à chaque instant la vraisemblance des observations $b_j(\mathbf{O}_t) \equiv P^{\Omega_t}(\mathbf{O}_t | s_j^t)$ (Eq. 5.30) pour chaque état singleton. Les vraisemblances sont ensuite interprétées comme des plausibilités conditionnelles [DS04], i.e.

$pl_b^{\Re^F}[s_j^t](\mathbf{O}_t) \equiv b_j(\mathbf{O}_t)$. Puis, nous pouvons appliquer le Théorème de Bayes Généralisé (GBT) pour obtenir $pl_b^{\Omega_t}[\mathbf{O}_t](S_j^t)$ (noté aussi $pl_b^{\Omega_t}(S_j^t)$ par souci de simplification de l'écriture) afin d'en déduire les croyances sur les sous-ensembles d'états.

**Une méthode hybride** On suppose que les modèles d'observation sont des mélanges Gaussiens *probabilistes* mais que la matrice de transition et l'a priori sont crédibilistes. Par conséquent, on calcule $pl_b^{\Omega_t}(S_j^t)$ de la même manière que précédemment puis, on effectue les propagations *forward* et *backward* crédibilistes. La variable crédibiliste $q_\gamma$ est ensuite calculée (Eq. 5.46), transformée en distribution de probabilités pignistiques puis utilisée dans les équations d'estimation des paramètres des mélanges Gaussiens probabilistes (Eq. 5.33). Cette méthodologie a été utilisée avec succès par Vannoorenberghe et Smets dans [VS05].

**Vers le développement d'un algorithme d'apprentissage crédibiliste** Nous donnons dans ce paragraphe quelques pistes pour développer un algorithme d'apprentissage des modèles d'observations totalement crédibiliste :

- **Représentation d'états -** Un HMM fait parti des *modèles d'états* et donc l'estimation des paramètres dans le cas probabiliste fait intervenir des mécanismes d'identification de modèles d'états [Mur02]. Ainsi, si les méthodes d'identification probabilistes sont étendues au cas crédibiliste, il suffira d'appliquer ces méthodes pour estimer les paramètres d'un CrHMM. A ce sujet, des recherches ont été très récemment lancées par T. Denœux sur les *modèles d'états crédibilistes* au laboratoire Heudiasyc.

- **Algorithme EM crédibiliste -** L'identification probabiliste fait généralement appel à un algorithme EM (*Expectation-Maximization*) dont l'extension au cas de données d'apprentissage crédibilistes a été proposée par Vannoorenberghe et Smets [VS05]. Le problème de l'apprentissage des paramètres d'un CrHMM peut alors être posé comme un problème de maximisation de la vraisemblance (généralisée aux fonctions et qui correspond à une plausibilité [Sme93]) des observations conditionnellement aux paramètres des mélanges. Cette plausibilité peut être calculée à partir des méthodes d'inférence que nous avons développées et utilisées dans ce manuscrit.

## 5.4 Conclusion

Nous avons adapté une partie des mécanismes des modèles de Markov cachés (HMM), initialement développés dans le cadre de la théorie des probabilités, aux fonctions de croyance que nous avons appelés *modèles de Markov cachés crédibilistes* (CrHMM). L'adaptation a consisté à transposer les équations de base des HMM pour qu'elles puissent être appliquées aux fonctions de croyance. En particulier, les algorithmes *forward* et *backward*, reposant sur les travaux de Smets concernant le Théorème de Bayes Généralisé [Sme93] ont été exprimés avec les communalités pour permettre un parallèle avec les versions probabilistes.

Notre contribution tient à la fois dans la généralisation des principaux mécanismes des HMM, dans la proposition d'un critère d'inférence en ligne lors de l'utilisation de l'algorithme *forward* et

dans le développement d'un algorithme de Viterbi crédibiliste pour la reconnaissance de séquences. L'information de conflit a été particulièrement exploitée alors qu'habituellement elle est éliminée.

Une méthode a été proposée pour estimer la matrice de transition crédibiliste. Cependant, il est nécessaire de compléter l'étude sur l'apprentissage notamment des modèles d'observations permettant de générer les fonctions de croyance. Les variables utilisées dans les HMM probabilistes pour cette tâche ont été exprimées dans le cadre des fonctions de croyance mais les formules d'estimation des paramètres des mélanges Gaussiens nécessitent d'être dérivées formellement. Cette perspective sous-entend l'utilisation probable de l'algorithme CrEM [VS05], extension de l'algorithme EM aux fonctions de croyance. Une autre perspective concerne l'exploitation du conflit à la fois pour l'estimation du nombre d'états et l'adaptation de la topologie des séquences. Enfin, il pourrait être intéressant de comparer formellement les CrHMM aux HMM possibilistes [MG00] car les fonctions de croyance généralisent les mesures de possibilités.

# Chapitre 6

# Reconnaissance d'actions et d'activités humaines dans des vidéos d'athlétisme

Ce chapitre est consacré à la mise en œuvre des méthodes proposées précédemment sur une application réelle permettant d'illustrer les performances de chacune des approches : le filtre temporel crédibiliste (chapitre 3), l'ordonnanceur CSS (chapitre 4) et le modèle de Markov caché crédibiliste (chapitre 5). L'application concerne la reconnaissance d'actions et d'activités humaines dans les vidéos d'athlétisme. La première section permet de situer et justifier le choix de cette application par rapport à la littérature (Fig. 6.1).

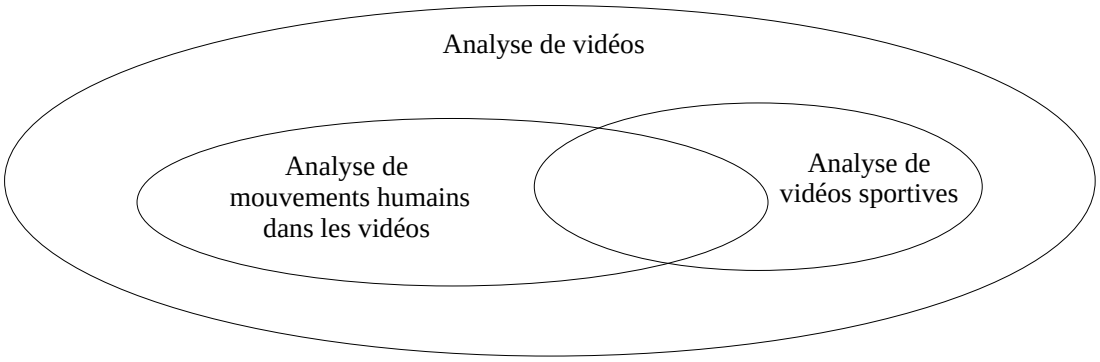



**FIG. 6.1 :** *Contexte de l'application.*

## 6.1 Introduction

Le nombre de chaînes radio-télévisées proposées aux téléspectateurs est de plus en plus important et un phénomène similaire apparaît sur l'Internet et la téléphonie mobile. En parallèle, de nombreux appareils électroniques à bon marché et à grande capacité de stockage sont disponibles permettant aux téléspectateurs d'enregistrer les événements qui les intéressent. Cependant, il existe peu d'appareils intégrant des capacités de tri facilitant l'*archivage* et surtout la *navigation* dans ces bases de vidéos personnelles ou publiques [OND+05]. Se pose donc le problème de retrouver la portion de vidéo correspondant à une requête d'un utilisateur [VT00, LSE02, XZT+06].

De nombreuses méthodes de découpage en *scène* et en *plan* ont été développées afin de pré-segmenter les vidéos en portions d'intérêt. Des informations sont ensuite extraites des images afin de décrire le *contenu* [VT00, LSE02, ICLB06, XZT+06] de ces segments. Classiquement, les descripteurs de *couleur* (comme les histogrammes [LHV03]), de *forme* (comme la géométrie d'objets ou les histogrammes de gradients spatio-temporels locaux [LP07]), de *mouvement* (par exemple les paramètres du mouvement de caméra [OB97]) et de *texture* (représentée par exemple par des coefficients de transformées d'ondelettes de Gabor [SB06]) sont utilisées [Gui06]. Une base de vidéos « modèle » peut ainsi être représentée par un ensemble de descripteurs. A partir de ces modèles, il est possible de construire un système de *recherche par l'exemple* : l'utilisateur émet une réquête haut niveau sur le type de vidéos qu'il recherche, par exemple « retrouver les portions de vidéo ou des personnes sont en train de boire » [LP07], et le système doit être capable de retranscrire cette requête en terme de relations entre les descripteurs puis de trouver les portions de vidéos en concordance avec la requête (problème du *fossé sémantique* [LHGT04]). La mise en correspondance se fait généralement à l'aide d'un calcul de distance locale ou globale [LHV03] entre, d'une part, les modèles de descripteurs et, d'autre part, des descripteurs calculés sur des nouvelles vidéos. Des boucles de retour (*feedbacks*) de l'utilisateur permettent par ailleurs de mettre à jour les descripteurs ainsi que le mécanisme de recherche [LSE02].

De nombreux systèmes dédiés ont été proposés car il est difficile de créer un système générique applicable à n'importe quel type de vidéos due à la variété de ces dernières. Nous nous focalisons en particulier sur les *vidéos de sports* car les retransmissions d'événements sportifs prennent une part importante dans les émissions retransmises à la télévision, sur l'Internet et, plus récemment, sur les téléphones mobiles [WYX05, RD06, TNB07]. De plus, les activités sportives suivent des règles [LHGT04] qu'il est possible d'utiliser lors du traitement.

### 6.1.1 Analyse des vidéos de sports

Les vidéos de sports représentent une proportion non négligeable des bases de vidéos et les enjeux liés à l'analyse automatique de ces vidéos sont cruciaux pour les professionnels, par exemple pour l'annotation [BBN04, MCJ+05, RCKL07]. La plupart des travaux de recherche se sont focalisés sur l'analyse des vidéos de sports diffusées à la télévision (*broadcast TV*). La première raison est que celles-ci véhiculent différents types d'informations pertinentes pour leur analyse comme les *paroles* des commentateurs, les indications *textuelles* données par les chaînes et les données *visuelles* représentées par les images. La seconde raison est qu'il existe de nombreuses applications autres que la *gestion des bases de vidéos* comme l'*analyse de performances et des tactiques* [She06] afin de donner des statistiques aux téléspectateurs, joueurs et entraineurs [YHYC05, BJA07], la *réalité augmentée* [KKO03, YYHL04, She06] afin de créer des environnements 3D pour visualiser des gestes techniques ou des fautes, et le *suivi d'objets* comme les balles de tennis ou les ballons de football [YLXT06, YLL+03, BJA07]. Mais, sans nul doute, l'application majeure reste la *création de résumés*.

Résumer une vidéo, sportive ou non, consiste à créer une nouvelle vidéo plus courte que la vidéo originale et contenant les informations essentielles. Il existe des résumés *statiques*, basés sur collection d'images clés, et *dynamiques*, formés de segments de vidéos (par exemple pour une bande annonce de film [ICLB06]). Dans la littérature, les termes *abstraction* [BKOK04] et *summarization* [SLS03, LEPS04, YLSL07] sont généralement employés pour décrire ce type de traitement.

Le résumé est important dans les vidéos de sports car les phases de jeu sont souvent entrecoupées de pauses publicitaires ou d'arrêts de jeu comme les fautes [BLZ+07, WCX+07] qu'un utilisateur ne souhaite généralement pas voir ni enregistrer [RDO06]. Les méthodes de résumé sont souvent multimodales [JS05] (voir aussi [PR03] qui se focalise sur la reconnaissance d'états affectifs, [Smi03] sur la sécurité et [DGH+02] sur l'analyse de gestes et d'expressions faciales dans les vidéos) et exploitent les informations visuelles, auditives et textuelles contenues dans les vidéos retransmises [CZKA96, WXC06, CH06, XZT+06]. L'audio permet par exemple de détecter des changements d'activité des spectateurs [Han03] comme des sifflements lors d'une faute ou des applaudissements lors d'un point marqué [TCP04] tandis que l'information textuelle apparaît par exemple pour donner un temps record ou un score [BLZ+07]. Un grand nombre d'informations est tirée de l'information visuelle dépendant de l'application [LWH03, Gui06].

Afin de résumer une vidéo, une première approche consiste à déterminer les instants de début et de fin des périodes d'intérêt. Pour cela, des auteurs ont proposé d'utiliser les émotions des sportifs [WCX+06] ou encore l'information sonore [RDO06]. Une autre approche, plus répandue, consiste à *détecter des événements* particuliers et intéressants pour l'utilisateur. Dans la littérature, on parle généralement de *highlights / event detection* [TCP04]. Par exemple, les auteurs de [XRDH05] proposent de résumer une vidéo en détectant tout d'abord des événements d'intérêt comme la courbure d'un joueur de golf prêt à swinguer, l'apparition des poteaux des cages lors d'un match de football ou encore le mouvement d'un batteur au baseball (voir aussi [RCKL07] pour la reconnaissance de gestes particuliers dans les vidéos de sports à basse résolution). Dans un deuxième temps, une analyse de la couleur et du mouvement de caméra permet de déterminer les segments intéressants autour des événements. D'autres méthodes sont basées sur la détection de ralentis ou de logos [HHH+07, DDHJ07], qui apparaissent généralement après un événement intéressant.

De nombreux sports ont fait l'objet de recherche[1] comme le football européen [ABC+03, SOMM04, BLZ+07, WCX+07], le football américain [LEPS04, DF07], le cricket [KS06], le volleyball [XMZY05], les jeux de raquettes [ZHX+07] comme le tennis de table [ZCDZ07] ou le tennis sur court [KGG+03, LMI06], le baseball [LWH03, LEPS04, CH06, LCL07], le basketball [ZXR+07], l'aérobic [BD01], le badmington [CZ07], le sumo [LEPS04], le snooker [DRK03], le patinage et l'athlétisme [Pir05, PBY06], la boxe [BCJ05, LMI06], la course de chevaux [OND+05, TCP04], la danse [EBMM03], le rugby [TCP04, BO05], le golf, le ski et le bowling [LWH03], le hockey [LL06], la natation [TDX+06], le plongeon, la gymnastique, le judo et le cyclisme [MCJ+05].

Les descripteurs et algorithmes de *bas niveau* utilisés pour l'indexation et le résumé des *vidéos d'athlétisme* sont généralement basés sur les informations de couleur *prédominante* (ex : le *vert* au football) et sur les amplitudes de mouvement de caméra [Gui06]. Nous proposons d'utiliser les mouvements des athlètes pour caractériser ces vidéos. Nous nous focalisons sur les épreuves où un seul athlète évolue, par exemple en *saut en hauteur*.

### 6.1.2 Analyse de mouvements humains

Alors qu'un être humain est capable de reconnaître des mouvements à partir de très peu d'information, par exemple avec la position des articulations [Joh75], il est beaucoup plus difficile de créer des systèmes de vision *artificiels* capables de telles prouesses, à part dans des conditions bien

[1] Les références citées dans ce paragraphe ont parfois des sports en commun.

particulières et maîtrisées [MG01]. Depuis les années 90, de très nombreux travaux visent à créer des systèmes permettant de répondre au moins partiellement à cette problématique de l'analyse de mouvements humains.

L'analyse de mouvements humains à partir de séquences d'images est important dans de nombreux domaines [Gra99, WHT03, MHK06], par exemple :

**Les interfaces Homme-Machine [PR03, JS05] –** Elles permettent d'interpréter des comportements humains dans le but de faire communiquer un homme et une machine. Les applications réelles sont par exemple la traduction automatique du langage des signes ou encore la reconnaissance de gestes. Les comportements humains sont estimés à partir de la fusion multimodale d'informations fournies par des capteurs visuels, auditifs, tactiles…

**La surveillance [HTWM04, Bre07] –** Les recherches en surveillance automatique ont explosé depuis les attentats de 2001 aux USA. L'objectif de ces systèmes est de détecter des comportements suspects [HHD00] par exemple dans des lieux publics [HBN00, Hon02], dans le métro [CABT04] ou les aéroports [XG03]. Des recherches récentes se sont portées sur la conception de systèmes à base de règles prédéfinies basées sur des ontologies [Hon02, HNB04, Vu04, Bre07], de systèmes de surveillance distribués coopératifs [VV05, MFS05] ou encore de systèmes de surveillance à domicile [Duc04, CGPV05, ZBTV07].

**L'analyse et la synthèse de mouvements [MHK06] –** Les applications sont multiples comme l'indexation de vidéos de sports, l'entrainement personnalisé [She06], les études cliniques de patients hospitalisés [Duc04] ou encore la création d'environnement virtuels [Pan06].

Le traitement peut être décomposé en trois étapes [WHT03] :

1. **L'extraction** de caractéristiques bas niveau et la détection de silhouettes ou des parties du corps humains. Cette première étape fait intervenir des techniques de séparation entre le fond et les objets d'intérêts et de segmentation [LHV03, FLB06] ainsi que des techniques d'extraction bas niveau comme les gradients spatio-temporels d'intensité [LMI06] à différentes échelles temporelles, l'analyse en ondelettes à différentes échelles spatiales [CC99] ou encore l'utilisation des descripteurs MPEG de mouvement [PCD02, MZ03].
2. **Le suivi de points** particuliers sur les silhouettes [WS03]. Cette seconde étape dépend de la précision recherchée lors du suivi. Par exemple, la synthèse de mouvements, utilisée au cinéma ou dans les jeux vidéos entre autres, nécessite généralement une modélisation 3D des personnes voire l'utilisation de marqueurs permettant une modélisation précise. Dans [GG04a], un modèle 3D de la sihouette d'un gymnaste est utilisé pour reconnaître des mouvements complexes. La méthode intègre un filtre à particules pour la mise en correspondance entre le modèle 3D et les images car le nombre de degrés de liberté est important. Cette méthode a été utilisée dans [GG04b] pour la reconnaissance biométrique [HTWM04]. Une méthode un peu moins précise et basée sur des modèles d'apparence a été proposée dans [RFZ07]. La méthode intégre un processus de mise à jour des modèles lors du suivi lui conférant une grande robustesse au changement d'illumination et au changement du fond des images. De très bonnes performances ont été obtenues sur des types de vidéos variées.
3. **L'inférence de comportements** [Mur02, WHT03, AP04]. Cette étape consiste à mettre en correspondance des modèles de comportements avec des observations. De nombreuses méthodes

existent et nous avons pu voir, dans les paragraphes précédents, quelques exemples basés sur des distances. Il existe par ailleurs des méthodes symboliques basées sur l'expression de règles de comportements à partir de données expertes. Ces règles sont formalisées par des langages objets, des opérateurs logiques et des machines à états finis [PB98, Vu04, Bre07]. Ces méthodes sont inspirées de l'algèbre de Allen [All83, AF94] et donc gèrent bien les contraintes temporelles. Cependant, il est difficile d'une part de prendre en compte l'incertitude sur les contraintes temporelles et, d'autre part de réaliser un apprentissage automatique [Mur95, Bre07]. Les méthodes les plus répandues pour reconnaître des mouvements restent les méthodes statistiques tels que les modèles de Markov Cachés (HMM : *Hidden Markov Models*) [Rab89], les réseaux Bayésiens Dynamiques (DBN : *Dynamic Bayesian Networks*) [Mur02] et quelquefois les réseaux de neurones [WHT03, MHK06] mais dans ce dernier cas, l'interprétation des modèles est difficile. Les HMM et les DBN sont des modèles probabilistes qui exploitent à la fois la puissance de la théorie des graphes pour la représentation et la programmation dynamique l'efficacité algorithmique et la rapidité des calculs. Ces modèles ont été très largement utilisés pour la reconnaissance de mouvements humains [LWH03, LL06] (voir aussi les récents états de l'art [WHT03, MHK06]) et de nombreuses variantes de ces méthodes existent [Mur02].

### 6.1.3 Contribution

Nous cherchons à analyser les mouvements d'un athlète dans les vidéos d'athlétisme. Pour la majorité des méthodes d'analyse de vidéos de sports, seules des informations très locales (comme les événements) ou globales (comme la segmentation en fonction du mouvement de caméra) sont utilisées. Nous proposons une analyse plus fine de ces vidéos en tentant de *reconnaître les mouvements des athlètes* grâce aux nouvelles méthodes de reconnaissance de séquences développées dans cette thèse. De plus, nous proposons un *comparatif* d'une part, entre plusieurs méthodes de *modélisation* de caractéristiques à l'aide de fonctions de croyance à partir desquelles nos algorithmes de reconnaissance seront appliqués, et d'autre part, entre nos méthodes « crédibilistes » et les HMM probabilistes. Enfin, nous présentons un système complet, de l'extraction à la reconnaissance en passant par la modélisation et le suivi de points, pour la reconnaissance de mouvements d'athlète dans les vidéos d'athlétisme.

L'une des difficultés majeures pour reconnaître des mouvements dans les vidéos d'athlétisme concerne le fait qu'elles sont acquises en caméra mobile [CPV04]. Dans la littérature, généralement, la caméra est statique et l'angle de vue est fixe, par exemple dans les travaux de Bobick et al. sur les modèles temporels [BD01] appliqués à la reconnaissance d'exercices aérobics. Quelques auteurs ont traité des vidéos en caméra mobile. Par exemple, Efros et al. [EBMM03] exploite le mouvement de la caméra pour la reconnaissance d'actions effectuées par des humains éloignés de la caméra (les silhouettes sont de taille $30 \times 30$ pixels). Pour cela, les auteurs utilisent un processus de mise en correspondance du flot optique sur des blocs d'imagettes avec des modèles pré-estimés. La méthode montre une bonne performance sur différents types de vidéos mais elle ne peut traiter des silhouettes de taille ou de forme différentes. Cette méthode a été améliorée dans [LMI06] avec la prise en compte de gradients spatio-temporels à plusieurs échelles de temps. Dans [YS05], les auteurs tentent de contourner le problème de la gestion du mouvement de caméra pour la reconnaissance d'action et l'indexation. Au lieu de modéliser les mouvements en fonction des angles de vue, ce qui s'avère une tâche très difficile, ils proposent d'utiliser la géométrie de la scène dans l'espace 3D+t. Ils projettent ainsi les caractéristiques extraites des vidéos dans ce nouvel espace et appliquent une mesure de similarité

pour la mise en correspondance avec des modèles de trajectoire pré-estimés. Les résultats montrent l'efficacité de cette méthode sur différents types de mouvements.

Dans le cadre de nos travaux, nous exploitons le mouvement de caméra comme une information pertinente sur le mouvement *global* de l'athlète. Pour cela, nous supposons que la caméra est mobile et suit l'athlète. Cette hypothèse est généralement vérifiée dans les meetings d'athlétisme où l'athlète est le centre d'intérêt.

Dans la section suivante, nous présentons le système de reconnaissance (Section 6.1). Puis, nous nous intéressons au problème de l'*extraction* de caractéristiques (Section 6.3) et à la *modélisation*, i.e. à la conversion des données numériques en croyances sur les actions effectuées par l'athlète (Section 6.4). Enfin, nous proposons une synthèse des *résultats* obtenus pendant cette thèse sur une base de $74$ vidéos que nous avons créée (Section 6.5).

## 6.2 Contexte expérimental

Dans cette section, nous présentons le système et décrivons la base de tests.

### 6.2.1 Architecture du système proposé

Nous avons mis en place un système (Fig. 6.2) permettant de reconnaître des actions et des activités humaines dans les vidéos d'athlétisme. Une *action* est interprétée comme un *état* tandis qu'une *activité* est représentée par une *séquence* d'actions.

Les actions et activités qui nous intéressent sont les suivantes :

- **Actions reconnues :** *courir*, *sauter*, *chuter* et *se relever*.
- **Activités reconnues :** *saut en hauteur*, *saut en longueur*, *triple saut* et *saut à la perche*.

L'entrée du système est une séquence d'images où l'athlète est debout à la première image. Les paramètres du mouvement sont en premier lieu estimés afin de détecter les objets mobiles. Ensuite, un algorithme de catégorisation nous permet d'estimer le nombre d'objets mobiles dans la vidéo et de déterminer si la vidéo concerne un sport individuel ou un sport collectif. Dans le cas du sport individuel, un algorithme de suivi de points est appliqué pour estimer la position de points particuliers sur la silhouette. La position de ces points nous permet de calculer un jeu de caractéristiques locales de mouvement de l'athlète. Les caractéristiques globales et locales obtenues pendant la phase d'*extraction* (Fig. 6.3) sont transformées en croyance sur les actions qui sont ensuite combinées dans le cadre du Modèle des Croyances Transférables. Les croyances générées à ce niveau représentent les entrées des algorithmes proposés dans cette thèse, à savoir le filtre TCF, l'ordonnanceur CSS et le modèle de Markov caché crédibiliste (CrHMM), qui permettent de reconnaître les actions et les activités de l'athlète.

### 6.2.2 Description de la base de vidéos

Le tableau 6.1 résume la base de vidéos utilisées en terme de nombre d'images par actions (*courir*, *sauter*, *chuter* et *se relever*) et le nombre de vidéos ($N_v$) par activités (*saut en hauteur*, *saut en longueur*, *saut à la perche* et *triple saut*).

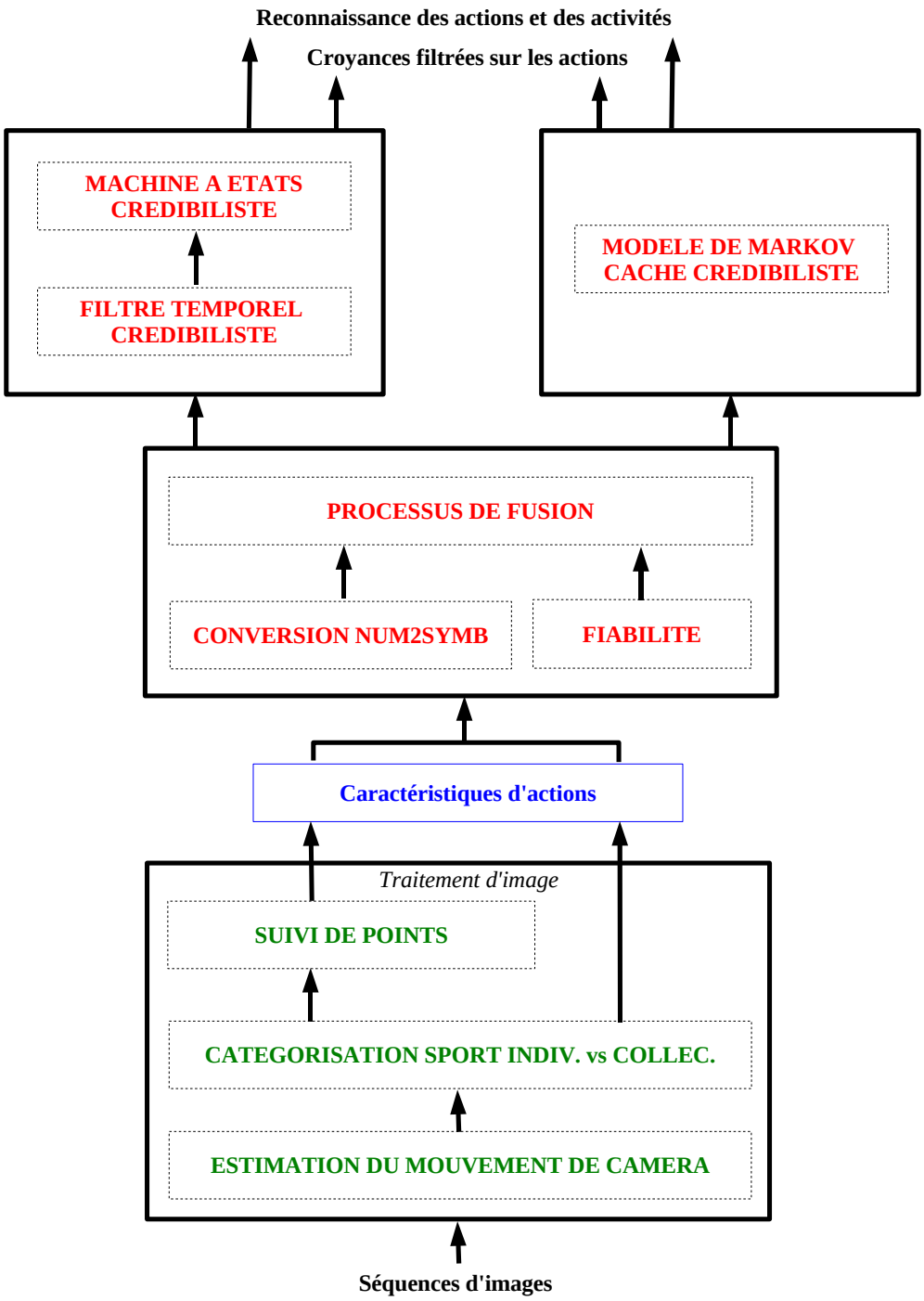



**FIG. 6.2 :** *Architecture du système.*

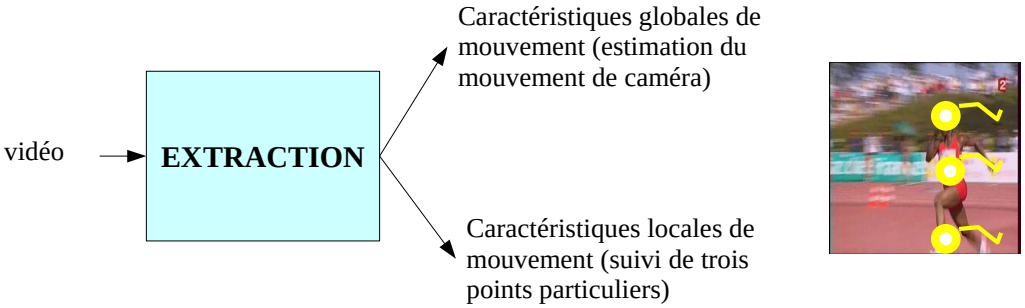



**FIG. 6.3 :** *Extraction d'informations sur les actions.*

Quelques images de la base de vidéos sont données sur la figure 6.4. La base est composée de 74 vidéos avec des mouvements de caméra différents, des conditions d'évolution des athlètes variées (extérieures en stade ouvert ou intérieures en stade fermé) et différents angles de vues. Lorsque l'axe optique de la caméra est parallèle au mouvement de l'athlète, il n'est pas possible de distinguer les actions à cause de la nature des caractéristiques extraites. On suppose donc que l'angle de vue de la caméra est proche de la perpendiculaire au mouvement de l'athlète. Les limites n'ont pas été évaluées précisément mais on estime que l'angle limite est d'environ $30^o$. Par exemple, sur la figure 6.4, l'image de la première ligne et première colonne montre un *saut en longueur* où l'angle est d'environ $70^o$ pour lequel aucun problème n'apparaît pour l'analyse, alors que les images aux lignes 1-colonne 3 et 3-colonne 1 montrent des *sauts à la perche* pour lesquels il est difficile d'extraire certaines caractéristiques.

| Saut/Action | $N_v$ | courir | sauter | chuter | se relever | Total |
|---|---|---|---|---|---|---|
| saut en hauteur | 15 | 1000 | 500 | 400 | 200 | 2100 |
| saut en longueur | 21 | 1800 | 500 | 700 | 600 | 3600 |
| saut à la perche | 26 | 2000 | 1400 | 1000 | 700 | 5100 |
| triple saut | 12 | 1200 | 400 | 600 | 400 | 2600 |
| Total | 74 | 6000 | 2800 | 2700 | 1900 | 13400 |

**TAB. 6.1 :** *Description de la base en terme de nombre d'images, où $N_v$ correspond au nombre de vidéos.*

## 6.3 Extraction

Les caractéristiques numériques pertinentes pour les actions que l'on cherche à reconnaître sont extraites de la séquence d'images.

### 6.3.1 Estimation du mouvement de caméra

Nous émettons l'hypothèse que la caméra suit l'athlète et cette hypothèse est vérifiée dans la majorité des vidéos d'athlétisme puisque l'athlète est le centre d'intérêt.

**FIG. 6.4 :** *Quelques images montrant l'hétérogénité de la base.*

Les paramètres du mouvement de caméra donnent donc une information sur le mouvement de l'athlète et on s'intéressera particulièrement aux mouvements suivants :

- **translation horizontale :** ce paramètre donne une information sur la véracité de l'action *courir* puisque au cours de cette action, la translation horizontale est généralement significative. De plus, au cours des actions *sauter*, *chuter* et *se relever*, la valeur de ce paramètre est généralement nulle.
- **translation verticale :** ce paramètre donne une information sur la véracité des actions *sauter*, *chuter* et *se relever*. Par exemple, une translation verticale positive sur plusieurs images consécutives signifie généralement que l'athlète saute ou se relève. La valeur de ce paramètre est en général nulle pour l'action *courir*.
- **zoom :** ce paramètre est utilisé comme une information complémentaire de la translation horizontale lorsque le mouvement de la caméra n'est pas parallèle au mouvement de l'athlète. De plus, le caméraman a généralement tendance à zoomer sur l'athlète lorsque celui-ci se relève.

Le mouvement de la caméra entre deux images sucessives est supposé linéaire et il peut être décrit par le modèle affine paramétrique suivant :

$$\begin{pmatrix} u(p_i) \\ v(p_i) \end{pmatrix} = \begin{pmatrix} F_{\text{hm}} \\ F_{\text{vm}} \end{pmatrix} + \begin{pmatrix} F_{\text{div}} & 0 \\ 0 & F_{\text{div}} \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} \tag{6.1}$$

où $[u(p_i), v(p_i)]^T$ est le vecteur de mouvement entre deux images successives du pixel $p_i$ situé aux coordonnées $(x_i, y_i)$ dans l'image, et $\{F_{\text{hm}}, F_{\text{vm}}, F_{\text{div}}\}$ sont les paramètres de translation horizontale, de translation verticale et de zoom respectivement.

L'estimation de ces paramètres est réalisée par le logiciel *motion2D* conçu à l'IRISA et s'appuyant sur la méthode d'Odobez et Bouthémy [OB95]. Cette méthode paramétrique consiste à déterminer les paramètres de mouvement minimisant la différence de déplacement entre deux images consécutives. L'estimation est itérative, multi-résolution et intègre un M-estimateur de Tukey robuste aux pixels aberrants pris en compte dans l'estimation. Cette méthode a été utilisée à de nombreuses reprises, par exemple pour l'indexation et le suivi de véhicules [PBPY02, PBY06, LMB02]. Un exemple de paramètres estimés est donné sur la figure 6.5 pour un *saut en hauteur*. Les barres verticales séparent les différentes actions à savoir *courir*, *sauter*, *chuter* et *se relever*.

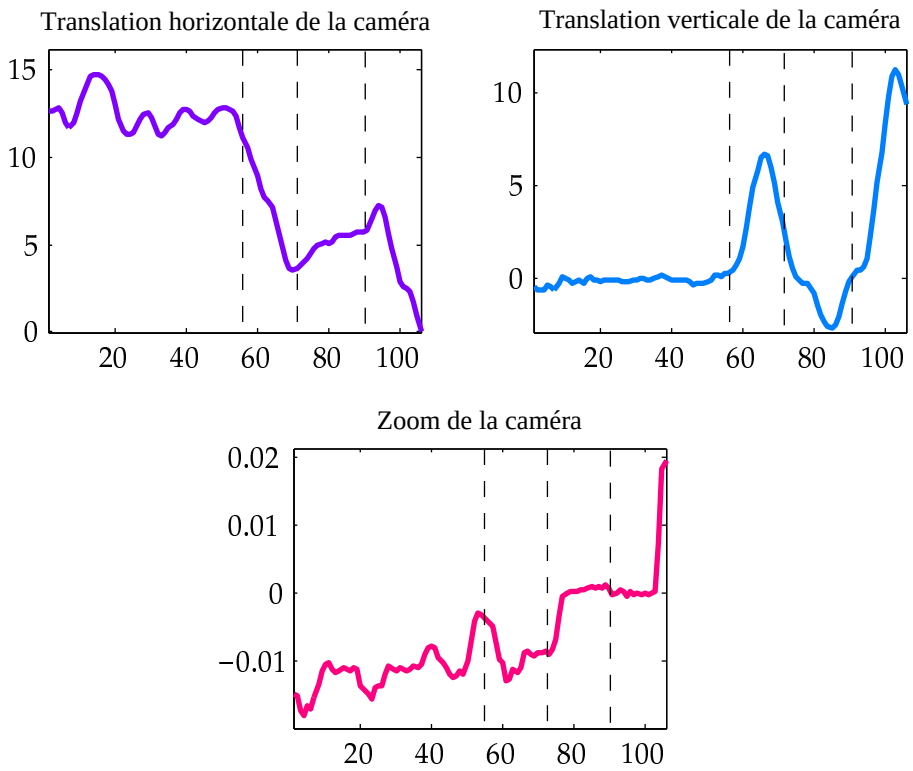



**FIG. 6.5 :** *Evolution des paramètres de translations horizontale et verticale (en pixels par image) et du zoom de la caméra estimés par* motion2D *pour un* saut en hauteur *avec 4 actions :* courir *sur* $[1, 58]$, sauter *sur* $[59, 72]$, chuter *sur* $[73, 90]$ *et* se relever *sur* $[91, 106]$.

L'estimation permet de générer une image en niveau de gris où l'intensité de chaque pixel dépend de son appartenance au mouvement dominant (Fig. 6.6-6.7 et 6.8, colonnes $4$ et $5$). Le mouvement dominant correspond au mouvement du fond de l'image dans le cas où la taille des objets mobiles est relativement faible par rapport au reste de l'image (empiriquement $30\%$, ce qui est vérifiée sur notre base de vidéos). L'athlète étant en mouvement par rapport au fond, il ne correspond pas au mouvement dominant et sa silhouette apparaît plus foncée.

**Robustesse de l'estimation du mouvement de caméra** La méthode d'estimation du mouvement de caméra est basée sur l'hypothèse que l'illumination est constante entre deux images [Bou04], cette contrainte étant relativement faible car *motion2D* intègre une compensation à ce niveau. L'autre contrainte concerne la taille des objets mobiles dont la proportion doit être faible par rapport au reste de l'image. Si leur taille est trop importante alors le *mouvement dominant* n'est plus celui de la caméra et par conséquent l'estimation des paramètres ne sera pas possible. Les images de mouvement dominant sont donc importantes et il faut s'assurer, sur la base de vidéos concernée, de la robustesse de l'estimateur qui les génère.

Les figures 6.6, 6.7 et 6.8 (colonnes $4$ et $5$) illustrent quelques unes des images de mouvement dominant. Plus un pixel est blanc plus il appartient au mouvement dominant.

La robustesse des estimateurs paramétriques dépend principalement de la variation de l'intensité lumineuse entre deux images successives (celles utilisées dans l'estimation) [Bou04]. Afin d'étudier cette variation nous avons réalisé les tests suivants :

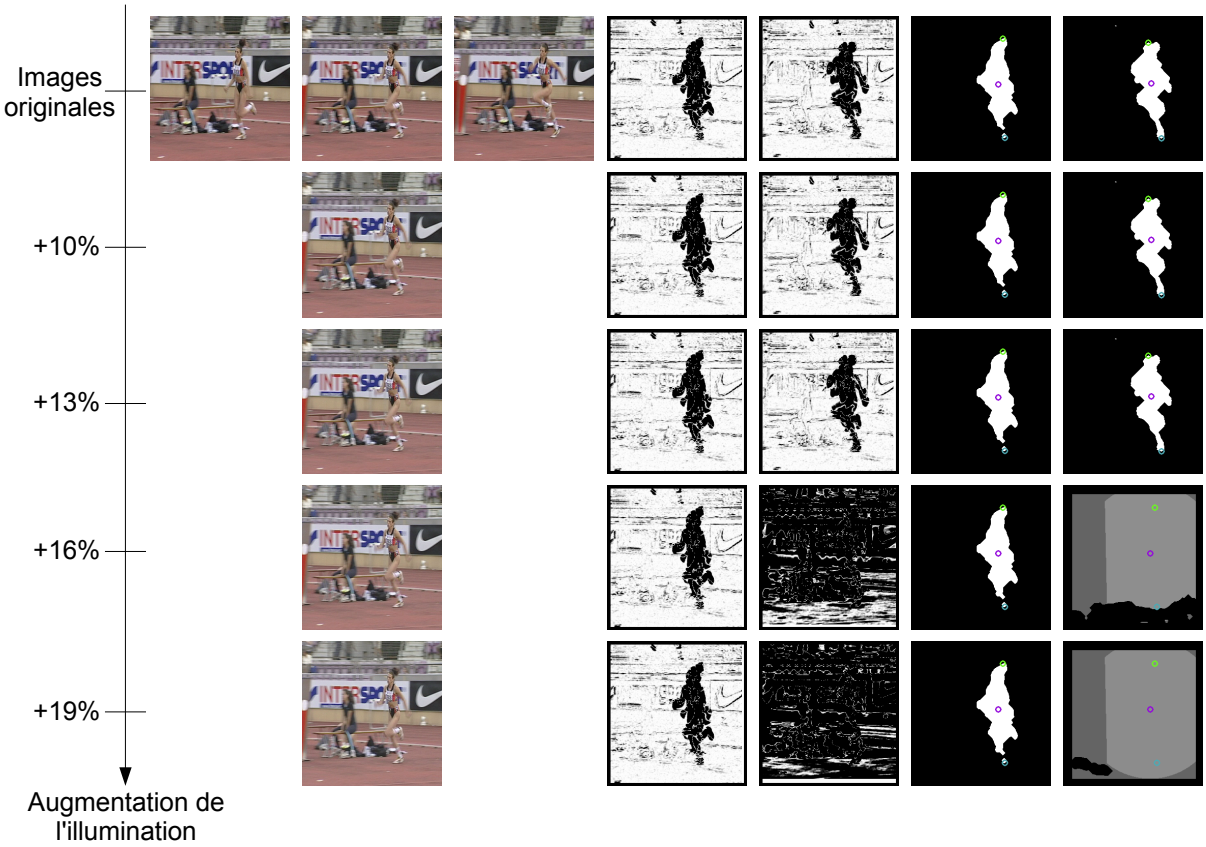



**FIG. 6.6 :** *Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en* extérieur *et obtenue à partir d'un site internet (moyenne qualité).*

1. Dans le premier test, nous déterminons la variation d'illumination maximale que peut supporter l'algorithme d'estimation *motion2D*.
2. Dans le second test, nous évaluons la variation d'illumination sur les séquences d'images de la base et comparons les résultats du second test à ceux du premier pour vérifier si les vidéos utilisées peuvent effectivement être traitées par *motion2D*.

Pour réaliser le premier test, nous considérons 3 vidéos caractéristiques de la base :

– une vidéo de *saut en hauteur* de *qualité moyenne* (en termes de flou de bougé et de compression) acquise en *conditions extérieures* obtenue à partir d'un *site internet*[2] (Fig. 6.6),
– une vidéo de *saut en hauteur* de *qualité faible* acquise en *conditions intérieures*, sous lumière artificielle, obtenue à partir d'un *enregistrement analogique* (Fig. 6.7),
– une vidéo de *saut à la perche* de *qualité élevée* acquise en *conditions extérieures*, sous lumière artificielle et au cours d'un meeting nocturne, obtenue à partir d'un DVD (Fig. 6.8).

Nous choisissons, dans chacune des trois séquences, trois images consécutives de qualité satisfaisante. Nous augmentons ensuite l'intensité moyenne sur l'image centrale $I(t)$ (colonne 2). Ainsi, une variation positive de l'intensité moyenne apparaît entre les deux premières images $I(t-1)$ et $I(t)$ (colonnes 1 et 2) et une variation négative entre les deux dernières images $I(t)$ et $I(t+1)$ (colonnes 2 et 3). Le taux de variation de l'intensité moyenne est calculé par un pourcentage de l'intensité moyenne

[2]Téléchargeable sur `www.dartfish.com`.

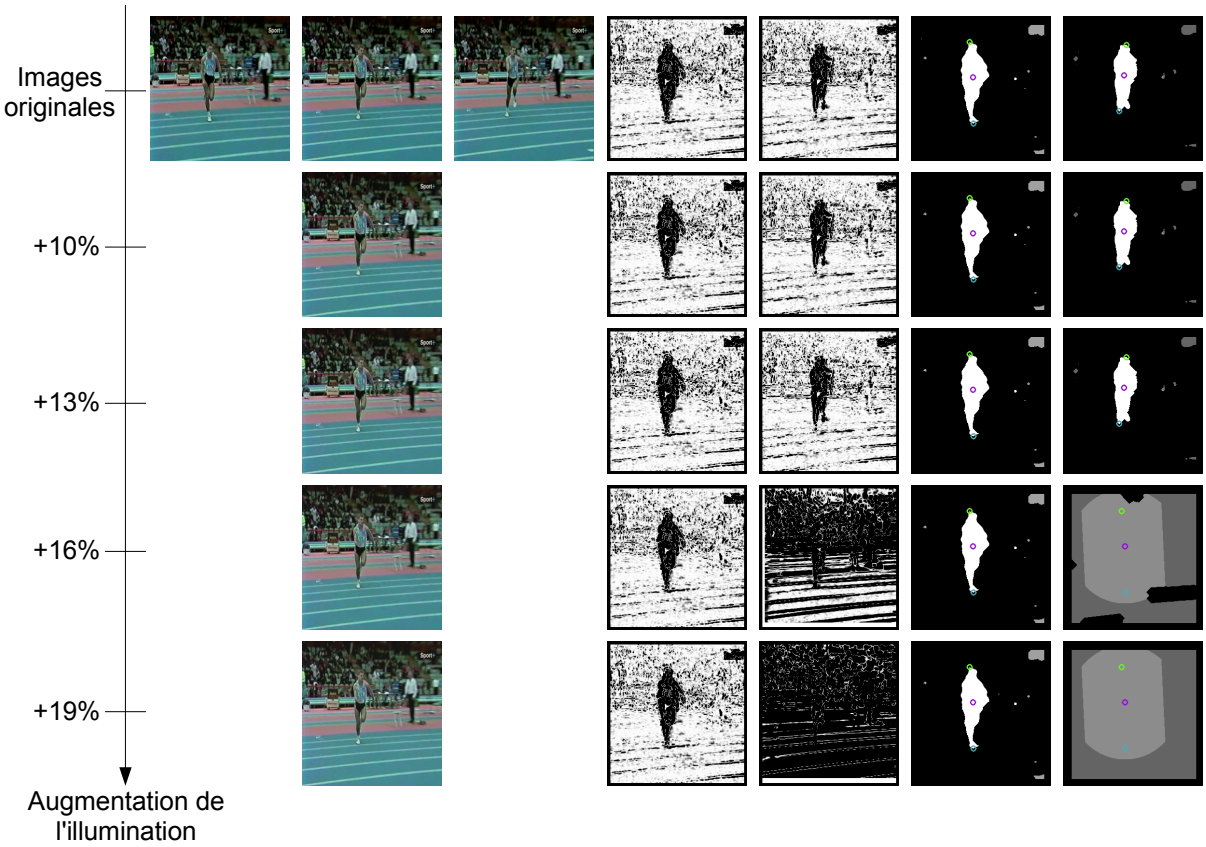



**FIG. 6.7 :** *Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en* intérieur *et obtenue à partir d'un enregistrement analogique (basse qualité).*

sur l'image centrale et la valeur du pourcentage est indiquée par la flèche à gauche des différentes figures.

Nous appliquons ensuite l'estimateur de mouvement de caméra menant à deux images de mouvement dominant : celle entre $I(t-1)$ et $I(t)$ et celle entre $I(t)$ et $I(t+1)$, chacune des estimations étant *perturbées* par un front montant et un front descendant de l'intensité moyenne.

Les résultats de l'estimateur sont donnés sur les figures 6.6-6.7 et 6.8 (colonnes $4$ et $5$). Ces résultats montrent que l'estimateur ne donne pas de bons résultats d'estimation lorsque la variation est supérieure à $13\%$ environ. On peut voir, qu'au delà de cette valeur, l'image de mouvement dominant est entièrement noire ce qui interdit tout traitement ultérieur.

La figure 6.9 illustre la variation moyenne d'intensité $100 \cdot (I(t) - I(t-1)/I(t-1))$ entre deux images successives (aux instants $t-1$ et $t$) pour un *saut en hauteur*, un *saut à la perche*, un *triple saut* et un *saut en longueur*. Les vidéos sélectionnées font parti des pires cas rencontrés dans la base. Comme le montre ces courbes, la variation reste bien inférieure à $13\%$, par conséquent l'hypothèse de variation relativement faible d'illumination est vérifiée sur la base permettant ainsi l'application de l'estimateur *motion2D*. Notons que nous avons développé une interface pour les tests de l'algorithme (Annexe 4).

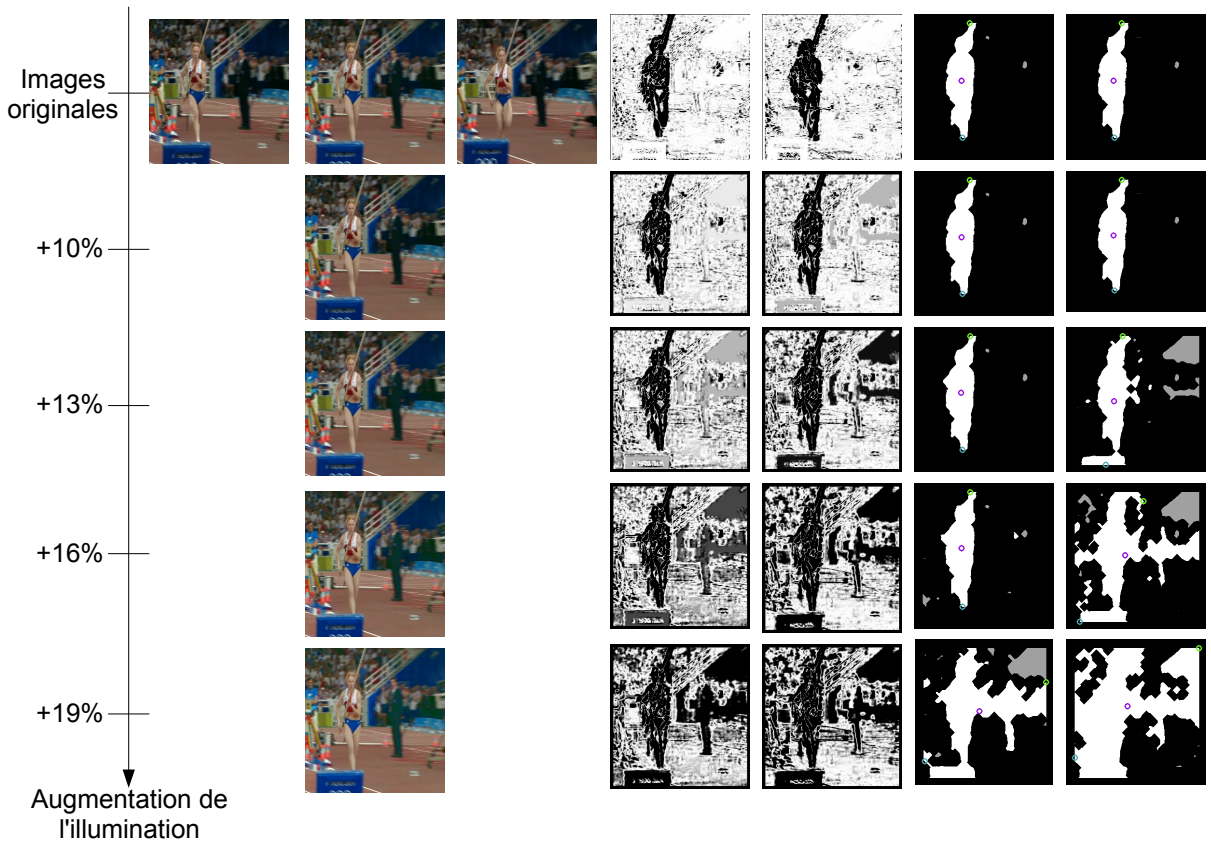



**FIG. 6.8 :** *Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en* extérieur *et obtenue à partir d'une DVD (haute qualité).*

### 6.3.2 Catégorisation de la vidéo : sport collectif vs. sport individuel

Avant de procéder au suivi de points (décrit dans le paragraphe suivant), nous cherchons à savoir si la vidéo en entrée du système contient un seul athlète en mouvement (Fig. 6.10). En collaboration avec l'université de Crète, nous avons mis en place un algorithme, décrit dans la thèse de Costas Panagiotakis [Pan06] ainsi que dans [PRT+07], permettant de distinguer les vidéos de sports *collectifs* (avec plusieurs athlètes mobiles) des vidéos de sports *individuels* (avec un seul athlète mobile). L'algorithme permet par ailleurs d'estimer le nombre d'objets mobiles. Ainsi, si l'algorithme indique que la vidéo correspond à un sport individuel alors le suivi de points (suivi de la reconnaissance des mouvements) est effectué.

L'algorithme de catégorisation repose sur les deux hypothèses suivantes : la taille des différents objets est similaire et, dans les vidéos de sports collectifs, les athlètes sont majoritairement en position debout. A partir de ces deux hypothèses, les objets de grande taille sont découpés en objets de taille égale à la taille moyenne des objets dans l'image.

L'algorithme de catégorisation est basé sur des caractéristiques de forme et sur des tables de règles *ad hoc* adaptées à l'application. Les étapes sont les suivantes :

- Estimation du mouvement de caméra comme expliqué précédemment. L'objectif est de détecter les objets mobiles.
- Binarisation : nous seuillons les images de mouvement dominant pour obtenir une image binaire où les objets mobiles apparaissent en blanc dans le fond noir. Puis, nous appliquons un filtre

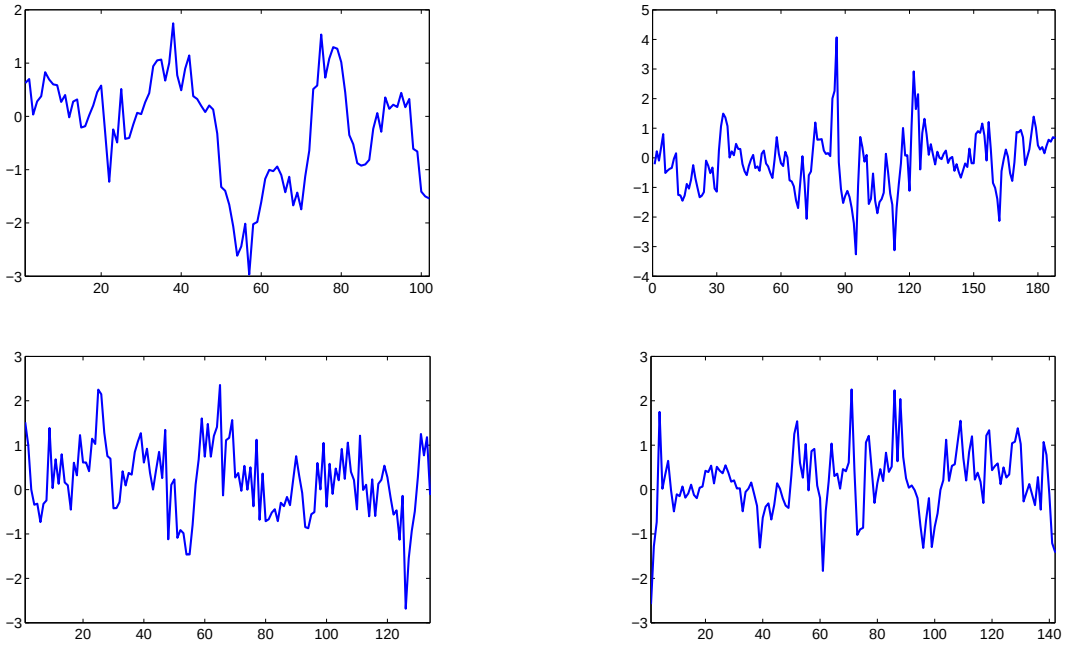


**FIG. 6.9 :** *Evolution en pourcent de la variation relative d'intensité lumineuse moyenne entre deux images successives dans (*haut–gauche*) un* saut en hauteur, *(*haut–droite*) un* saut à la perche, *(*bas–gauche*) un* triple saut *et (*bas–droite*)* un saut en longueur.

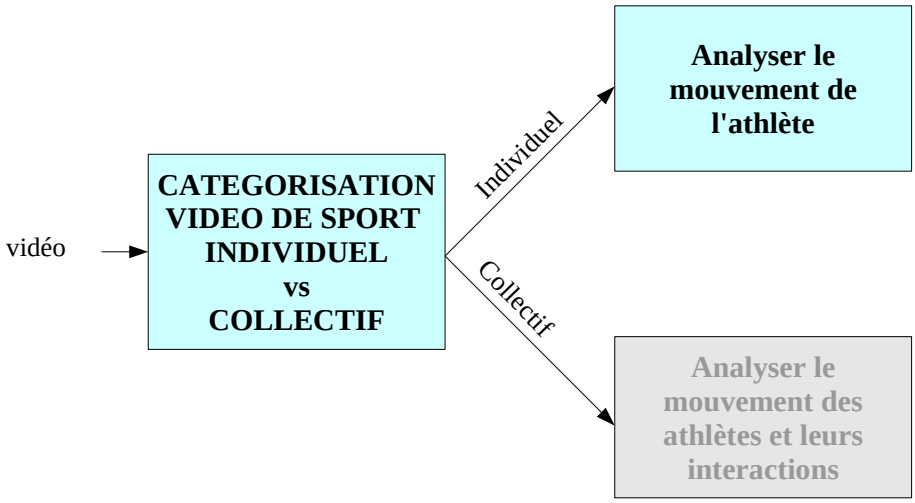



**FIG. 6.10 :** *Discrimination entre des vidéos de sports d'équipe et individuel.*

médian (noyau carré $7 \times 7$) et une ouverture morphologique (élément structurant carré de taille $7 \times 7$) afin de supprimer les petits objets et pour assurer une connexité des blocs de pixels formant la sihouette.

– Filtrage : un moyenneur sur 5 images successives est appliqué suivi d'un seuillage pour supprimer les objets furtifs.
– Estimation de caractéristiques de forme des objets : l'excentricité[3], la surface et l'angle entre

[3]L'excentricité d'une région de pixels est calculée en faisant correspondre une ellipse à cette région pour approximer

l'axe principal de l'objet et l'axe horizontal de l'image sont estimés pour chaque objet et à chaque image. Une conversion numérique-symbolique de ces caractéristiques permet d'obtenir une probabilité d'appartenance des objets à des symboles du type « la surface est importante » ou « l'excentricité est grande ». Ces probabilités sont combinées avec des tables de règles puis seuillées pour déterminer si un objet est ou non formé d'une ou plusieurs silhouettes.

L'algorithme estime le nombre d'objets dans chaque image et génère un critère de qualité qui est fonction de la valeur des probabilités utilisées dans la table de règles. Le nombre d'objets et le critère de qualité sont combinées afin de générer une fonction de croyance binaire (vidéo de sport individuel ou collectif) sur le type de vidéo.

Nous avons évalué cette méthode [PRT+07] sur une base de $252$ vidéos menant à un taux de bonne classification d'environ $97\%$. Un exemple d'application est donné sur la figure 6.11 pour un *triple saut* (sport individuel) et sur la figure 6.12 pour un sprint de 100m (sport collectif). Dans le cas du *triple saut*, on remarque qu'un deuxième athlète apparaît en arrière plan (le nombre d'objet est proche de 2 sur la courbe du dessous). Pour l'exemple du 100m, $8$ athlètes apparaissent au début de la séquence puis, à cause de la position de la caméra, un phénomène d'occlusion apparaît cachant $5$ athlète. Les figures 6.13 et 6.14 montrent le résultat de l'algorithme pour une vidéo non segmentée composée de plusieurs sports collectifs et d'un sport individuel. La masse de croyance (Fig. 6.13) sur le fait que la vidéo est un sport individuel permet de détecter précisemment le segment de vidéo ($[40, 100]$) correspondant au *saut en hauteur*, la courbe verte représente le doute apparaissant lors de la transition entre les différents sauts.

### 6.3.3 Suivi de points

Lorsque la vidéo a été classée en « sport individuel », le mouvement local de quatre points caractéristiques : la tête, le centre de gravité et les deux pieds de l'athlète, permet de discriminer les actions recherchées. L'algorithme développé par Costas Panagiotakis dans sa thèse [Pan06] et décrit dans [PRT+06] permet de détecter et suivre ces 4 points. Dans la suite du manuscrit, nous nous contenterons de $3$ en fusionnant les deux pieds par un opérateur moyenne (nous prenons la moyenne des coordonnées verticale et horizontale des deux pieds).

L'algorithme de détection et de suivi [PRT+06] requiert des images binaires générées lors de la catégorisation de vidéos (par seuillage des images de mouvement dominant).

La méthode de suivi est basée sur la géométrie de la silhouette et ne pas prend en compte l'information de couleur [WS03] qui se trouve être beaucoup trop variable dans le cas des vidéos d'athlétisme réelles. L'algorithme fonctionne en deux temps :

1. **Détection des points :**
   (a) On détermine la position du centre de gravité de la silhouette (qui peut être composée de plusieurs blocs de pixels non connexes) ainsi que l'axe d'inertie passant par le centre de gravité.
   (b) L'axe d'inertie traverse la silhouette en deux points dont l'un est la tête (tête en haut, par exemple lors d'une course, ou tête en bas, par exemple lors d'une chute). Les deux positions sont mémorisées et le choix est fait dans l'étape suivante.

sa forme (en calculant des moments du second ordre) puis en calculant le rapport entre le demi grand axe et le demi petit axe. De cette manière, l'excentricité tend vers l'unité lorsque l'ellipse tend vers le cercle et vers l'infini lorsque l'ellipse s'allonge.

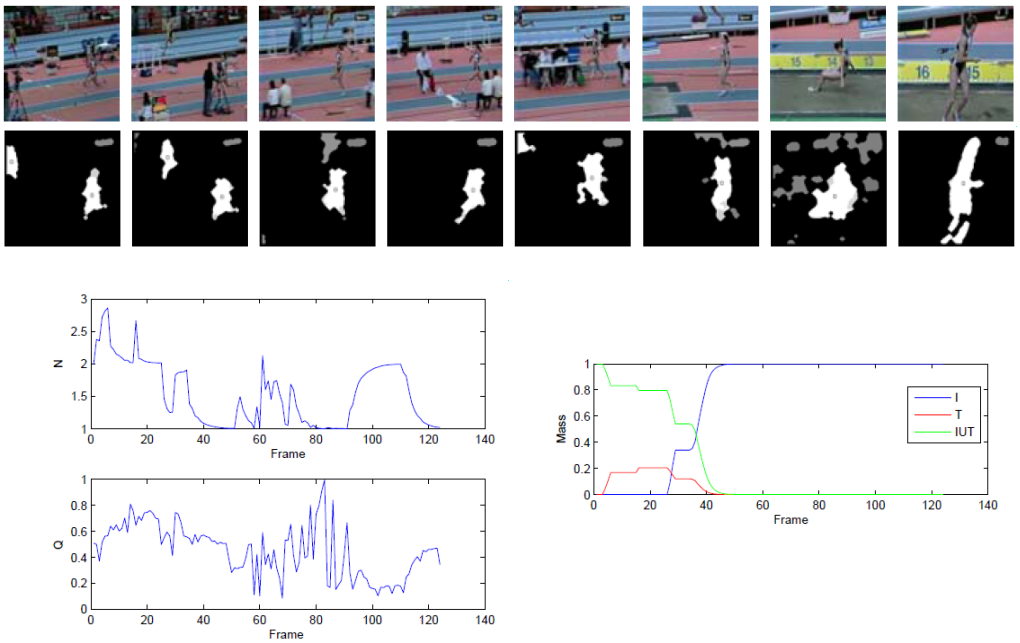



**FIG. 6.11 :** *Résultats de l'algorithme de catégorisation pour un* triple saut*, tiré de [PRT+07]. La première rangée d'images correspond à une partie des images originales, la seconde rangée représente le résultat de l'algorithme. La silhouette apparaît en blanc et son centre de gravité avec un petit carré. Les autres blocs de pixels (en gris) correspondent à du bruit (éliminé automatiquement par l'algorithme). La figure en bas à droite représente la masse de croyance sur le fait que la vidéo correspond à un sport collectif (T, en rouge), un sport individuel (I, en bleu) et le doute entre ces deux hypothèses (en vert). Ici la vidéo est un* triple saut *correctement classé en tant que sport individuel. Les figures en bas à gauche correspondent au nombre d'objets (valeur approximative réelle dont seule la partie entière a un sens) dans la vidéo (première courbe) et à un indice de qualité concernant l'estimation (deuxième courbe).*

(c) Pour les deux cas de l'étape 2 (tête en haut et tête en bas), la position des pieds est calculée. Le premier pied est choisi comme le point le plus éloigné du centre du gravité et situé vers le bas si la tête est en haut (vers le haut si la tête est en bas). La position du second pied est calculée par maximisation de l'aire du triangle isocèle formé par le centre de gravité, le premier pied et le second pied dont la position est recherchée.

2. **Suivi des points :** On sélectionne les quatre points qui sont les plus proches des quatre points à l'instant précédent en calculant la distance euclidienne entre les positions à $t-1$ et $t$ pour les deux solutions puis en minimisant cette distance.

Les figures 6.15 donnent quelques exemples de suivi de points pour les quatre types d'activités.

### 6.3.4 Les caractéristiques de mouvement utilisées

La position des points n'est pas assez robuste pour être utilisée comme une caractéristique car la position de l'athlète varie beaucoup dans une même vidéo. A partir de la position de points, nous calculons deux caractéristiques discriminantes :

**La variation verticale de la position du centre de gravité ($F_{vcg}$) –** Nous utilisons l'algorithme de détection de tendance développé par Charbonnier [Cha05]. Cet algorithme calcule l'erreur entre

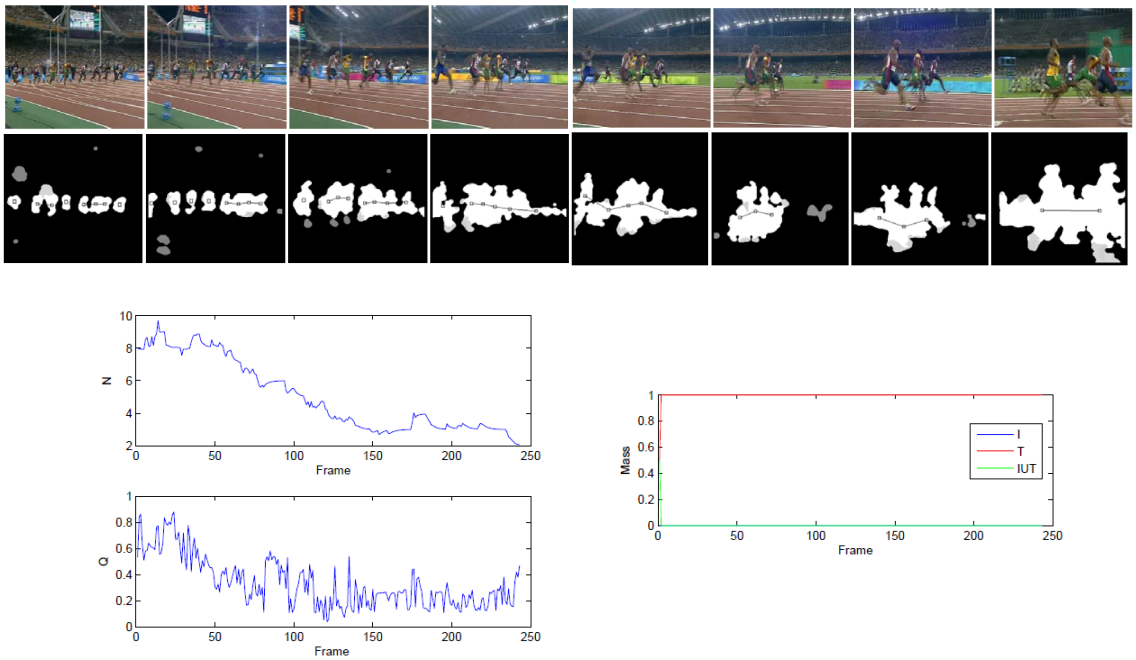



**FIG. 6.12 :** *Résultats de l'algorithme de catégorisation pour un sprint (100m), tiré de [PRT+07]. Voir la légende de la figure 6.11. Ici la différence vient du fait que la vidéo correspond à un sport collectif. Au départ de la course, les athlètes sont en ligne et la caméra de face donc les* 8 *sprinters sont bien détectés. Puis la caméra bouge et son axe optique devient perpendiculaire au mouvement des athlètes. A cause des occultations, seulement deux ou trois athlètes sont détectés.*

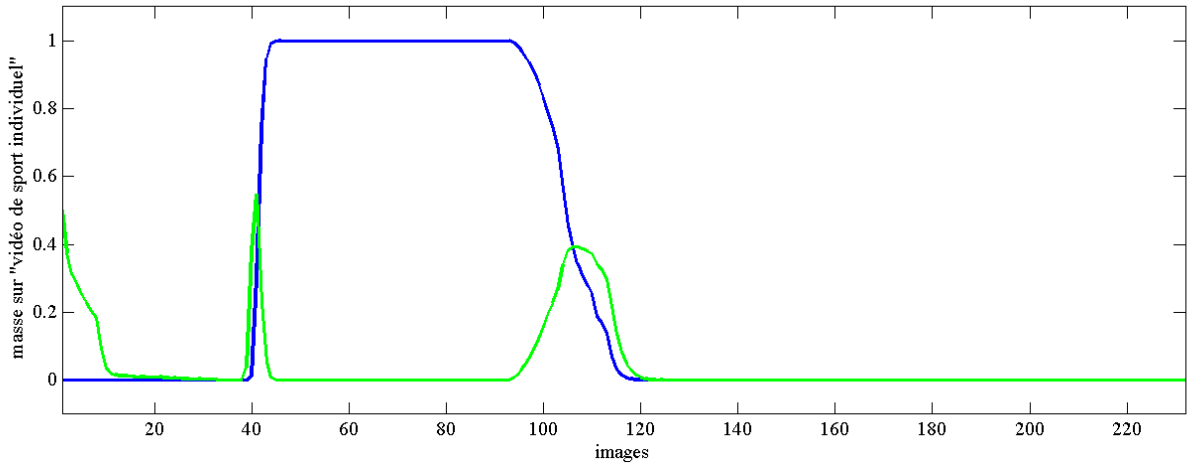



**FIG. 6.13 :** *Evolution de la masse de croyance sur le fait que la vidéo est un sport individuel fournie par l'algorithme de catégorisation pour une vidéo composée de plusieurs sports collectifs et d'un sport individuel (*saut en hauteur*). L'algorithme fonctionne en ligne et il est capable d'estimer le nombre d'objets mobiles (ici des athlètes) dans la vidéo au fil de l'eau. La zone détectée (de l'image* 40 *à l'image* 100 *environ) correspond au* saut en hauteur*, la courbe verte représente les transitions entre les sports. Les images originales et les images de la détection sont représentées sur la figure 6.14.*

un modèle linéaire d'évolution de la caractéristique et les valeurs de la caractéristique considérée. Lorsque l'erreur cumulée est importante (on utilise un seuil d'arrêt), le modèle est réestimé

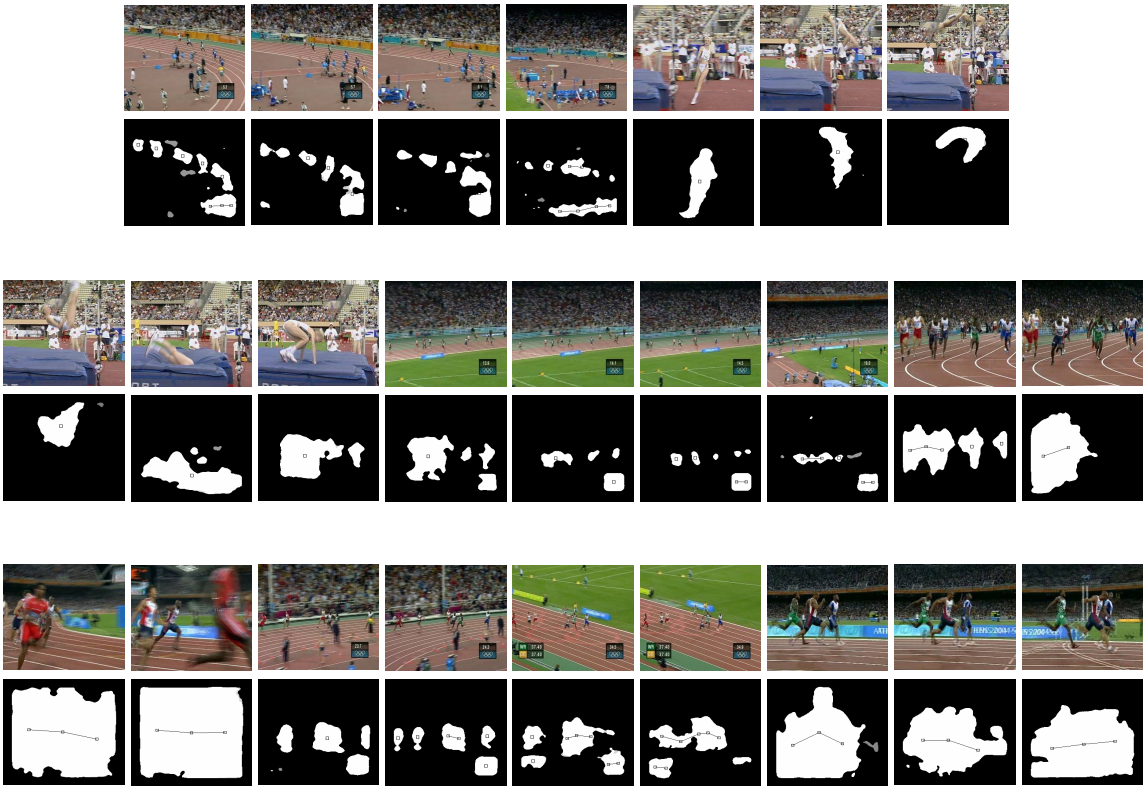


**FIG. 6.14 :** *Images originales et images de la détection de l'algorithme de catégorisation pour une vidéo composée de plusieurs sports collectifs et d'un sport individuel (*saut en hauteur*). Cette figure est associée à la courbe de la figure 6.13.*

sur un intervalle de temps de quelques images. Nous considérons l'ensemble des coefficients directeurs comme une caractéristique. Un exemple est donné sur la figure 6.16 pour un *saut en hauteur*.

**L'angle *swing*** $F_{swg}$ **–** L'angle entre l'axe principal de la silhouette et l'axe horizontal de l'image est appelé « angle swing » qui nous donne une information sur la position de l'athlète dans l'espace. L'axe principal de l'athlète est donné par la droite passant au plus près des trois points de la silhouette. Un exemple est donné sur la figure 6.17 pour un *saut en longueur*.

Le vecteur de caractéristiques est donc formé de cinq éléments : les deux paramètres précédents et les trois paramètres de mouvement de caméra. On a donc, à chaque instant, le vecteur de caractéristiques observées suivant :

$$\mathbf{O}_t = [F_{hm} \quad F_{vm} \quad F_{div} \quad F_{vcg} \quad F_{swg}] \tag{6.2}$$

Le problème est à présent de lier ces observations aux actions et activités à reconnaître.

Par ailleurs, on calcule automatiquement deux coefficients de fiabilité à chaque image : un relatif à la qualité de l'estimation du mouvement de caméra et un autre concernant le suivi de points. Le premier coefficient, noté $\alpha_{sup}$ repose sur l'hypothèse que le nombre de pixels appartenant à la silhouette varie peu entre deux instants. Si ce n'est pas le cas alors l'estimateur est supposé non fiable. Le coefficient est calculée en deux étapes : on estime d'abord la différence entre le nombre de pixels

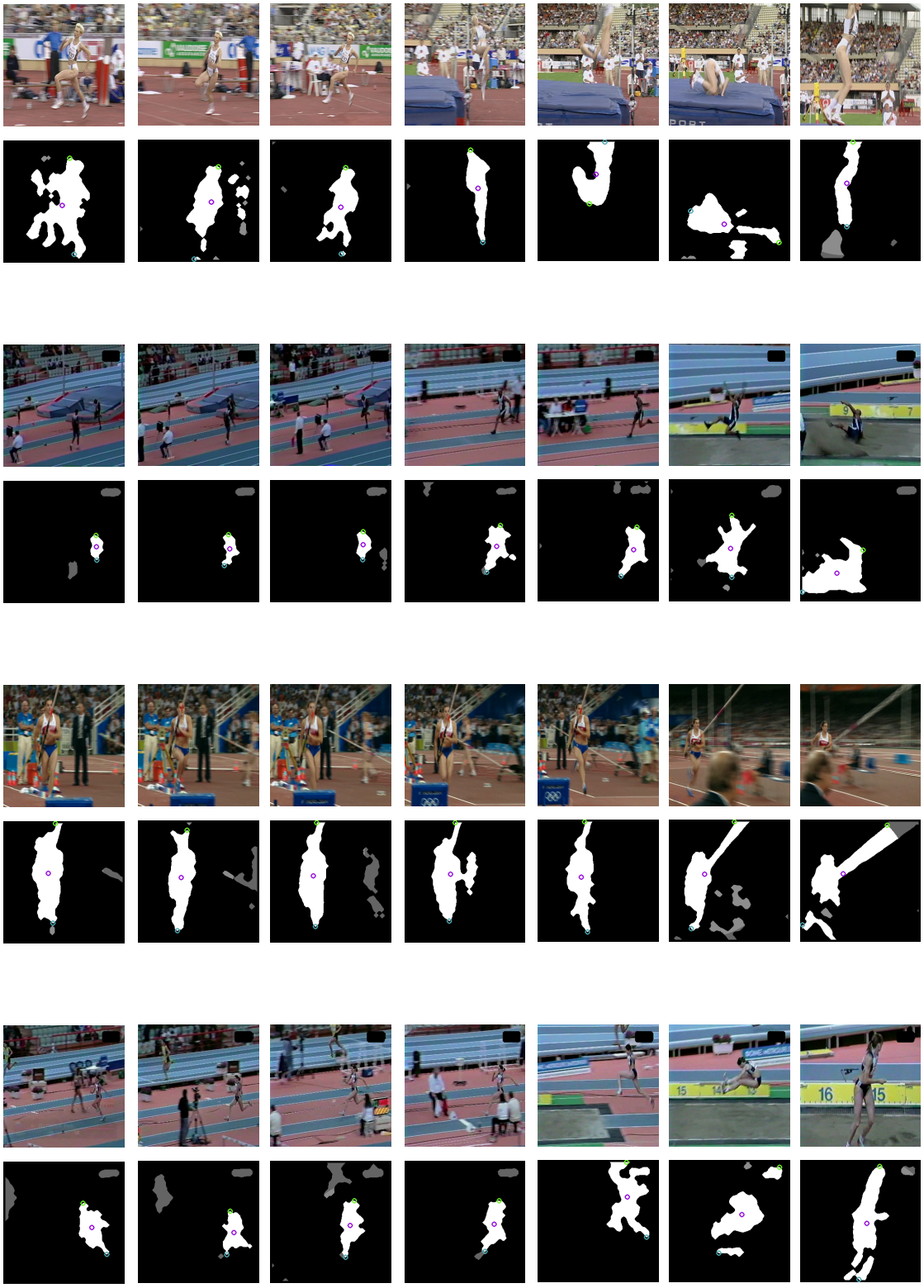


**FIG. 6.15 :** *Résultat du suivi de points pour un* saut en hauteur*, un* saut en longueur*, un* saut à la perche *et un* triple saut.

de la silhouette à un instant $t$ et le nombre de pixels de la silhouette à $t-1$. La différence est supposée être proche de $0$ avec un écart-type de $3$. La valeur de la différence est donc transformée en fiabilité $\alpha_{sup} \in [0,1]$ à l'aide d'une Gaussienne $\mathcal{N}(0,3)$.

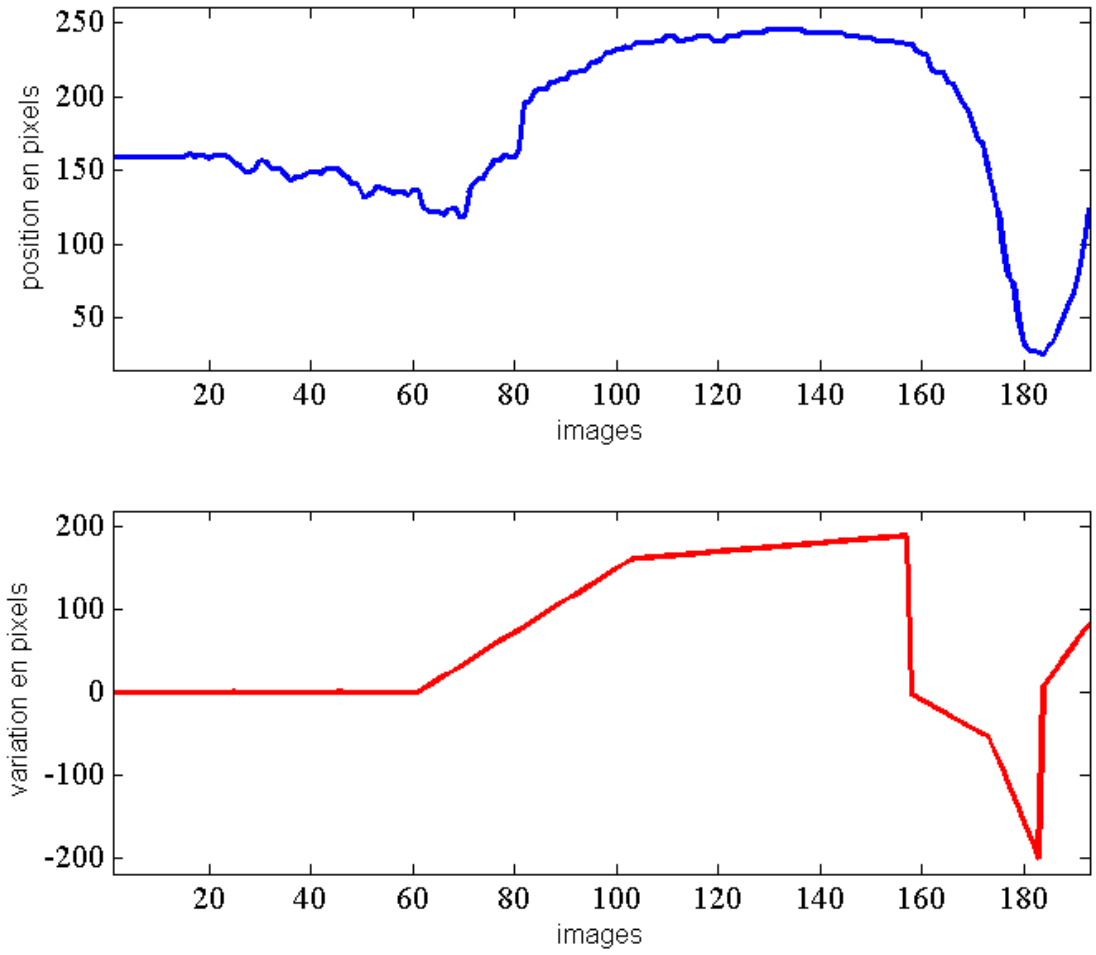



**FIG. 6.16 :** *Illustration de la variation verticale du centre de gravité. En haut est représentée l'évolution au cours du temps de la position (en pixels) verticale du centre de gravité. Les paliers sur la courbe du bas correspondent aux tendances de variation estimés par des modèles linéaires (valeur des coefficients directeurs). Exemple : le centre de gravité a une variation verticale positive importante sur l'intervalle* $[60, 100]$.

Le second coefficient, noté $\alpha_{tck}$ repose sur l'hypothèse que la distance entre la tête et le centre de gravité à un instant $t$ est similaire à cette même distance à l'instant précédent $t-1$. Si ce n'est pas le cas, alors le suivi est supposé non fiable. Le coefficient est calculé en prenant l'opposé de la valeur absolue de la différence entre ces deux distances. Le coefficient de fiabilité est à valeur dans $[0, 1]$ car les distances ont été normalisées par rapport à la taille de l'image.

## 6.4 Modélisation

Nous présentons les modèles de conversion liant les observations numériques $\mathbf{O}_t$, extraites à partir des vidéos, aux croyances sur les actions. Cette étape de modélisation est cruciale pour obtenir des performances correctes et ce quel que soit le formalisme employé (probabilité, possibilités ou fonctions de croyance) et elle est dépendante de l'application. Nous proposons plusieurs méthodes de modélisation afin de détecter les actions : une première méthode est paramétrique, une seconde est non-paramétrique et la troisième est basée sur la connaissance experte.

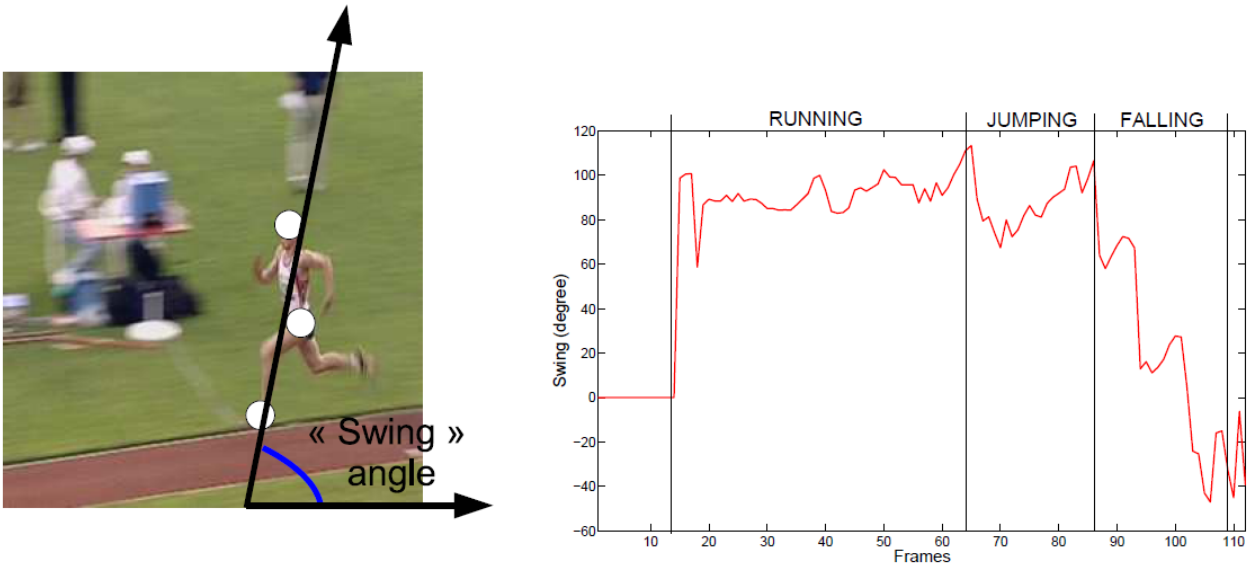



**FIG. 6.17 :** *Principe du calcul de l'angle et un exemple d'évolution pour un* saut en longueur*, tiré de [RPRP07].*

### 6.4.1 Problème posé

Nous cherchons à définir une fonction de croyance sur la véracité des actions pour un jeu de caractéristiques $\mathbf{O}_t$ donné à un instant $t$ (on fait l'hypothèse qu'une image est disponible à chaque instant $t$). Dans le cas du filtre temporel TCF et de l'ordonnanceur CSS, une fonction de croyance binaire doit être générée pour chaque action à chaque instant. Pour une action $s_j^t$ $j \in \{courir, sauter, chuter, se\ relever\}$, le cadre de discernement est :

$$\Omega_t^j = \{T_t^j, F_t^j\} \tag{6.3}$$

Dans le cas du CrHMM, la croyance doit être définie sur l'ensemble des actions et donc le cadre de discernement est :

$$\Omega_t^s = \{s_1^t, s_2^t, s_3^t, s_4^t\} \tag{6.4}$$

Le problème d'obtention des fonctions de croyance à partir des observations $\mathbf{O}_t$ peut être vu comme un problème de *reconnaissance des formes* [DS06, Quo06]. Dans ce contexte, on cherche à construire une *fonction* de l'espace des observations (aussi appelées *attributs*), $\Re^F$, vers l'espace des actions (représentant des *classes*), $\Omega_t^j$ ou $\Omega_t^s$. Cette fonction est appelée *classifieur* de type III lorsqu'une fonction de croyance est générée [Quo06]. Un classifieur $\mathcal{C}$ est défini de la manière suivante :

$$\begin{array}{rcccl} \mathcal{C} & : & \Re^F & \rightarrow & \Omega \\ & & \mathbf{O}_t & \mapsto & m^{\Omega} \end{array} \tag{6.5}$$

où $F$ est la dimension de $\mathbf{O}_t$ et $\Omega$ représente soit $\Omega_t^j$ soit $\Omega_t^s$.

Dans la littérature, de nombreuses méthodes ont été proposées pour estimer cette fonction. Ces méthodes peuvent être regroupées en trois catégories [DS06] :

– **Estimation directe** de la croyance a posteriori sur l'ensemble des classes (e.g. les réseaux de neurones [Den00], les arbres de décision [EMS01, VD02]).

– **Estimation de la croyance a posteriori avec le Théorème de Bayes Généralisé** à partir de densités conditionnelles aux classes et d'un a priori sur les classes [App91, VD01, DS04, CCFM04, RS04, RS05a, DS06]. Dans cette famille, il est nécessaire de connaître le modèle mathématique des densités conditionnelles.

– **Estimation non-paramétrique** de la croyance à partir d'un ensemble d'observations « proches » de l'observation à classer $\mathbf{O}_t$ (e.g. le K-plus proches voisins évidentiels utilisant le modèle de distance de Denœux [Den95, ZD98a, ZD98b, Den00, DZ01, VD02]). L'estimation non-paramétrique est intéressante lorsque les modèles des classes ne sont pas connus.

Nous nous sommes focalisés sur les deux derniers types de méthodes.

### 6.4.2 Vraisemblance et Théorème de Bayes Généralisé

On suppose disposer d'un modèle d'action pour chacune des actions $s_j^t$ (Fig. 6.18). Le schéma de classification suit les étapes suivantes :

1. Estimation des densités de probabilités générant une vraisemblance $P(\mathbf{O}_t|s_j^t)$ pour chaque action à chaque instant. On suppose que les densités sont des mélanges Gaussiens et nous appliquons un algorithme EM (Espérance-Maximisation) permettant d'estimer la moyenne et la variance de chaque Gaussienne. Pour l'estimation, nous avons appliqué le protocole suivant :
   – Pour chaque action, l'ensemble d'apprentissage correspond à $30\%$ de la base. Nous avons réalisé plusieurs tests afin de sélectionner les vidéos formant l'ensemble d'apprentissage en cherchant à avoir un taux de classification d'au moins $60\%$ sur l'ensemble de la base.
   – Le nombre de Gaussiennes[4] est, respectivement, pour les actions *courir* (état $s_1^t$), *sauter* (état $s_2^t$), *chuter* (état $s_3^t$) et *se relever* (état $s_4^t$) de $10$, $4$, $4$ et $8$.

2. A chaque instant $t$, et comme proposé par plusieurs auteurs dont Smets [DS04, Sme93, App91], nous assimilons la vraisemblance $P(\mathbf{O}_t|s_j^t)$ à la plausibilité $pl^{\Re^F}[s_j^t](\mathbf{O}_t)$ et nous appliquons le Théorème de Bayes Généralisé (Eq. 2.24c). Lorsque le cadre de discernement correspond à $\Omega_t^s$, on a :
$$m^{\Omega_t^s}[\mathbf{O}_t](S^t) = \prod_{s_j^t \in S^t} pl^{\Re^F}[s_j^t](\mathbf{O}_t) \cdot \prod_{s_j^t \in S^t} \left(1 - pl^{\Re^F}[s_j^t](\mathbf{O}_t)\right) \tag{6.6}$$
   Lorsque le cadre de discernement concerne une action particulière, i.e. $\Omega_t^j$ (cas binaire), comme requis dans l'ordonnanceur CSS, il est nécessaire de redéfinir les croyances de $\Omega_t^s$ vers $\Omega_t^j$. Pour cela, on effectue un grossissement $\Omega_t^s \rightarrow \Omega_t^j$ comme expliqué dans le chapitre 3-§ 3.6.2 (l'idée du grossissement est de transférer la masse sur l'action considérée vers l'hypothèse $T_t^j$, i.e. l'action est vraie, et la masse sur les autres actions est transférée sur $F_t^j$, i.e. l'action est fausse). On obtient alors une BBA binaire $m^{\Omega_t^j}[\mathbf{O}_t]$ pour chaque action.

Les croyances binaires $m^{\Omega_t^j}[\mathbf{O}_t]$ sont traitées par l'ordonnanceur CSS, qui intègre le filtre TCF, pour reconnaître à la fois les actions et les activités. La BBA $m^{\Omega_t^s}[\mathbf{O}_t]$ sur l'ensemble des actions est traitée par le HMM crédibiliste pour reconnaître là aussi les actions et les activités.

[4] Le nombre de Gaussiennes a été déterminé en utilisant un critère proposé dans [FJ02].

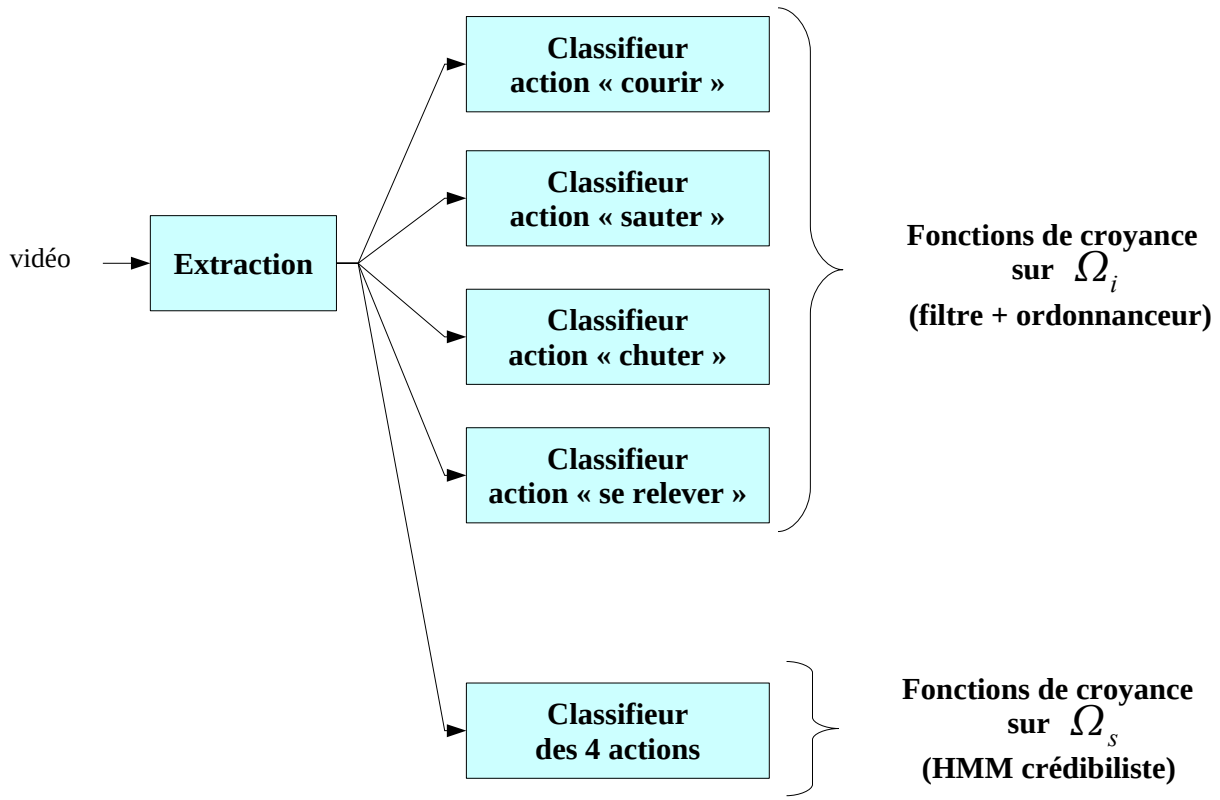



**FIG. 6.18 :** *Deux modélisations possibles : soit un classifieur par action ou un classifieur pour les quatre actions. Le premier cas est utilisé pour les tests sur l'ordonnanceur CSS et le deuxième cas pour le CrHMM.*

### 6.4.3 Le modèle de distance

On suppose disposer d'un ensemble d'apprentissage $\mathcal{L} = \{\mathbf{O}_n, m_n^{\Omega^s}\}$ où $n$ est un indice d'échantillon. Chaque échantillon $e_n$ est composé d'observations $\mathbf{O}_n$ étiquetées par une fonction de croyance $m_n^{\Omega^s}$ définie sur l'ensemble des actions $\Omega^s$ (on omet volontairement l'indice de temps ici). Si la classe $s_j \in \Omega^s$ de l'échantillon $e_n$ est connue précisément alors la fonction de croyance est *catégorique* ($m_n^{\Omega^s}(s_j) = 1, s_j \in \Omega^s$). Au contraire, si la classe est totalement inconnue alors la fonction de croyance *vide* est associée à l'échantillon ($m_n^{\Omega^s}(\Omega^s) = 1$). Dans notre application, l'étiquetage s'avère intéressant pour les transitions entre les actions, imprécises par nature, auxquelles on associera une croyance vide.

Pour une nouvelle observation $\mathbf{O}_t$, on cherche la BBA $m^{\Omega_t^s}[\mathbf{O}_t]$ reflétant le type d'action. Dans le modèle de distance de Denœux [Den95, ZD98a, ZD98b, Den00, DZ01, VD02], la croyance qu'une nouvelle observation $\mathbf{O}_t$ appartienne à l'une des classes (actions) de $\Omega_s^t$ est donnée par la combinaison conjonctive des BBA des $K$ plus proches voisins (déterminés par la distance Euclidienne) de l'observation $\mathbf{O}_t$ pondérées par leur distance à $\mathbf{O}_t$. Les étapes sont les suivantes :

1. **Recherche des $K$ plus proches voisins** en utilisant la distance Euclidienne entre les échantillons de $\mathcal{L} = \{\mathbf{O}_n, m_n^{\Omega^s}\}$ et l'observation courante $\mathbf{O}_t$. Le résultat est un ensemble de $K$ échantillons $\mathcal{L}_k = \{\mathbf{O}_k, m_k\}$.

2. **Calcul de la BBA** $m_k^{\Omega_t^s}[\mathbf{O}_t]$ pour chaque échantillon $k$ dans $\mathcal{L}_k$ :

$$\begin{aligned} m_k^{\Omega_t^s}[\mathbf{O}_k](\{s_j^t\}) &= \alpha \cdot \phi_q(\delta(\mathbf{O}_k, \mathbf{O}_t)) \\ m_k^{\Omega_t^s}[\mathbf{O}_k](\Omega_t^s) &= 1 - \alpha \cdot \phi_q(\delta(\mathbf{O}_k, \mathbf{O}_t)) \\ m_k^{\Omega_t^s}[\mathbf{O}_k](B) &= 0, \quad B \in 2^{\Omega_t^s} \backslash \{\Omega_t^s, \{s_j^t\}\} \end{aligned} \tag{6.7}$$

où $\phi_q(\delta(\mathbf{O}_k, \mathbf{O}_t)) = \exp(-\gamma_q \cdot \delta(\mathbf{O}_k, \mathbf{O}_t))$, $\delta(\mathbf{O}_k, \mathbf{O}_t)$ est la distance Euclidienne entre $\mathbf{O}_k$ et $\mathbf{O}_t$, $\alpha \in ]0, 1]$ et $\gamma_q \geq 0$.

3. **Combinaison des** $K$ **BBA pour estimer** $m^{\Omega_t^s}[\mathbf{O}_t]$ avec la combinaison conjonctive normalisée :

$$m^{\Omega_t^s}[\mathbf{O}_t] = m_1^{\Omega_t^s}[\mathbf{O}_1] \oplus \cdots \oplus m_k^{\Omega_t^s}[\mathbf{O}_k] \oplus \cdots \oplus m_K^{\Omega_t^s}[\mathbf{O}_K] \tag{6.8}$$

Comme dans le cas précédent, si le cadre de discernement concerne une action particulière, i.e. $\Omega_t^j$ (cas binaire), on effectue un grossissement $\Omega_t^s \rightarrow \Omega_t^j$ comme expliqué dans le chapitre 3-§ 3.6.2.

Le problème avec ce type de modèle concerne le fait que la masse de croyance est généralement affectée en grande partie aux singletons (le modèle génére une BBA quasi-Bayésienne). Pour remédier à ce problème, nous proposons d'utiliser un affaiblissement généralisé de la BBA générée par ce modèle afin de transférer de la masse sur les sous-ensembles d'hypothèses. Les coefficients de fiabilité sont ici choisis égaux à $0.1$ (valeur empirique). Nous avons choisi $K = 5$ et $\alpha = 0.99$ de manière empirique. La valeur de $\gamma_q$ a été optimisée[5] avec la méthode décrite dans [Den95].

### 6.4.4 Modélisation *experte*

Les conversions numériques-symboliques des caractéristiques en BBA peuvent être réalisées via une modélisation inspirée des sous-ensembles flous. Ce type d'approche est particulièrement adaptée lorsque trop *peu de données* d'apprentissage sont disponibles pour une approche statistique et lorsque les caractéristiques ainsi que les éléments du cadre de discernement sont *interprétables* par un humain.

La conversion est réalisée par des trapèzes comme illustré sur la figure 6.19 dont les $4$ seuils sont fixés à l'aide de connaissances expertes sur l'application. Une croyance est obtenue à chaque instant et pour chaque caractéristique et un mécanisme de fusion basé sur des tables de règles permet d'obtenir une croyance sur les différentes actions.

Ce type de méthodes a été utilisé pour l'analyse de postures humaines [GCBR05b, GBC05, GCB06, Gir06], pour l'analyse d'expressions faciales [Ham07, Ham05b, Ham05a] et pour la reconnaissance d'actions et d'activités humaines [RPRP07].

Dans le cadre de notre application, les règles pour chaque action suivent un schéma en deux niveaux [PRT+06] : d'abord une détection grossière des actions pour déterminer si l'athlète court, saute ou chute. Puis, une détection raffinée permet de discriminer les actions au sein des activités.

La détection grossière des actions se fait de la manière suivante :

**courir –** SI l'athlète est debout ($F_{swg}$ proche de $90^o$) ET SI un mouvement de translation de la caméra apparaît ($F_{hm}$) OU SI un zoom *arrière* apparaît ($F_{div}$) ALORS l'action *courir* est vraie.

**sauter –** SI une variation verticale positive de la position du centre de gravité apparaît ($F_{vcg}$) OU SI un mouvement vertical positif de la caméra apparaît ($F_{vm}$) ALORS l'action *sauter* est vraie.

[5] Code Matlab disponible sur `http://www.hds.utc.fr/~tdenoeux/`.

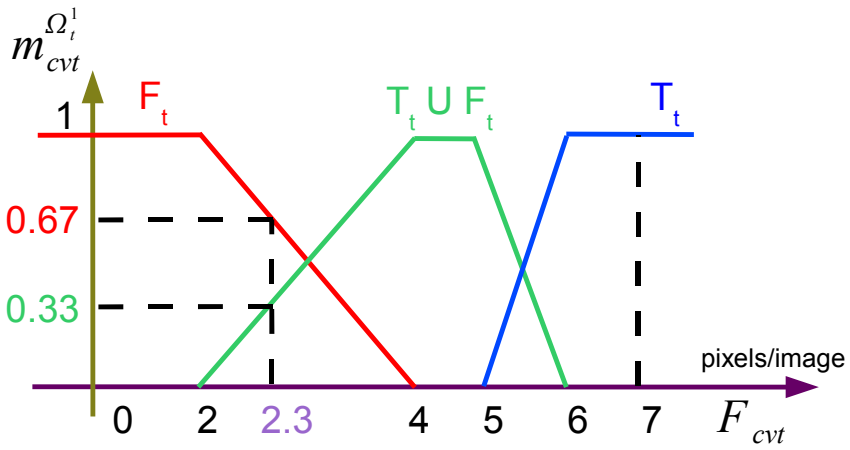



**FIG. 6.19 :** *Conversion numérique-symbolique inspirée des sous-ensembles flous : la caractéristique de translation verticale de la caméra ($F_{cvt}$) est transformée en BBA sur l'action* courir, $m^{\Omega_t^1}_{cvt}$, *définie sur* $\Omega_t^1 = \{T_t^1, F_t^1\}$.

**chuter –** SI une variation verticale négative de la position du centre de gravité apparaît ($F_{vcg}$) OU SI un mouvement vertical négatif de la caméra apparaît ($F_{vm}$) ALORS l'action *chuter* est vraie.

**se relever –** SI une variation verticale positive de la position du centre de gravité apparaît ($F_{vcg}$) OU SI un mouvement vertical positif de la caméra apparaît ($F_{vm}$) ET SI un zoom *avant* apparaît ($F_{div}$) ALORS l'action *se relever* est vraie.

Les seuils des trapèzes utilisés pour traduire les valeurs numériques ($F_{vcg}$, $F_{hm}$, $F_{div}$, $F_{swg}$) en croyances sur les symboles associés aux variables sont fixés de façon *ad hoc* en partant d'histogrammes pour guider nos réglages. Les OU et les ET logiques utilisés dans les règles permettent de combiner les croyances. Par exemple la croyance sur l'hypothèse « l'action *se relever* est vraie » est calculée en multipliant la croyance sur le symbole *important*, obtenue à partir de la valeur numérique de $F_{vcg}$, avec la croyance sur le symbole *important*, obtenue à partir de la valeur numérique de $F_{div}$.

Ensuite, pour distinguer les actions *sauter* et *chuter* dans les différentes activités, nous étudions l'évolution de l'angle. En effet, quelle que soit l'activité, ces actions impliquent une variation verticale de la position du centre de gravité mais la position de l'athlète dans l'espace est dicriminante. Pour illustration, la variation théorique de l'angle pour un *saut en hauteur*, un *saut à la perche* et *un saut en longueur* est donnée à la figure 6.20 pour le cas où l'athlète bouge de gauche à droite.

Après le traitement par l'ordonnanceur, on connait les intervalles temporels où les actions ont été détectées. Sur ces intervalles et à partir de la figure 6.20, on détecte les événements discriminants suivants :

**E**$_1$ si l'angle est entre $0^o$ et $45^o$ pendant l'action *sauter*, cette action est étiquetée comme appartenant à un *saut en hauteur*.

**E**$_2$ si l'angle est entre $-115^o$ et $-75^o$ pendant l'action *chuter*, cette action est étiquetée comme appartenant à un *saut en hauteur*.

**E**$_3$ si l'angle est entre $-115^o$ et $-75^o$ pendant l'action *sauter*, cette action est étiquetée comme appartenant à un *saut à la perche*.

**E**$_4$ si l'angle est entre $75^o$ et $115^o$ pendant l'action *chuter*, cette action est étiquetée comme appartenant à un *saut à la perche*.

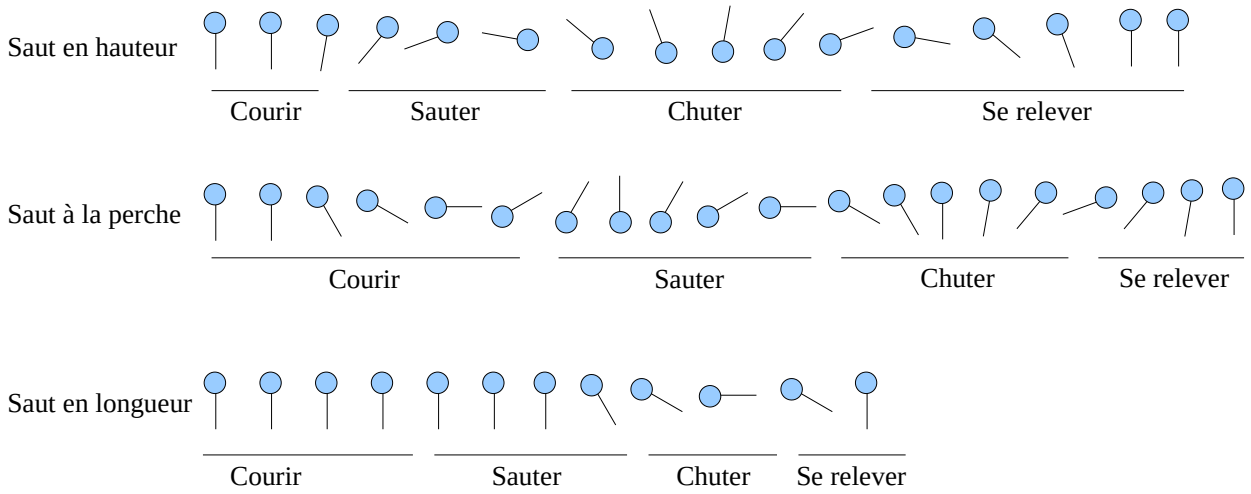



**FIG. 6.20 :** *Evolution théorique de l'angle pour un* saut en hauteur*, un* saut à la perche *et un* saut en longueur *pour un mouvement de caméra de gauche à droite.*

**E**$_5$ si l'angle est entre $135^o$ et $180^o$ pendant l'action *chuter* et que $E_3$ n'est pas apparu alors cette action est étiquetée comme appartenant à un *saut en longueur*.

Les intervalles numériques de valeurs de l'angle considérés dans ces règles, par exemple $[130^o, 180^o]$ sont stricts car on cherche à prendre une décision sur le type d'action.

Ces événements sont définis pour un mouvement de caméra allant de gauche à droite. Ce mouvement est détecté par l'événement suivant :

**E**$_6$ si la moyenne de la translation horizontale est négative pendant l'action *courir* alors le mouvement de la caméra (assimilé à celui de l'athlète) va de gauche à droite (et vice-versa).

Des événements similaires sont créés pour le cas ou le mouvement de caméra va de droite à gauche.

On dispose par ailleurs d'un détecteur de perche (l'objet utilisé par le perchiste) qui a été développé dans [PRT+06] et qui s'est avéré très robuste ($23/26$ détections correctes) dans le cadre de notre base de vidéos. L'algorithme consiste à détecter un objet long et fin rattaché à la silhouette et un seuillage de l'excentricité de cet objet nous permet de décider si une perche apparaît effectivement dans la vidéo. On dispose donc d'un nouvel événement :

**E**$_7$ si un objet « perche » est détecté dans l'action *courir* alors la vidéo est un *saut à la perche*.

Cet événement nous permet de contraindre le suivi de points à suivre une évolution donnée (processus *feed-forward*). En effet, dans les vidéos de *saut à la perche*, l'athlète se met littéralement « en boule » lors de la montée ce qui rend difficile l'estimation de l'angle et a pour effet d'inverser la position des points lors du suivi. Si l'événement $E_7$ apparaît alors on force l'algorithme de suivi à placer la tête de l'athlète en dessous du centre de gravité lors du début de l'action *sauter*.

La détection d'un objet « perche » à la fin de l'action *sauter* ou au début de l'action *chuter* est généralement due au fait que l'athlète a raté son saut et qu'il a touché la barre. En effet, lorsque la barre est touchée, celle-ci est détectée comme un objet mobile. On peut donc détecter les *sauts à la perche* et les *sauts en hauteur* ratés.

| | CSS | CrHMM | HMM |
|---|---|---|---|
| EX | T1 | - | - |
| GMM | T2 | T3 | T5 |
| EDC | - | T4 | T6 |

**TAB. 6.2 :** *Conditions expérimentales : association des méthodes de modélisation et des algorithmes de reconnaissance.*

## 6.5 Evaluation

La méthode symbolique (ordonnanceur CSS) et la méthode statistique (CrHMM) développées dans cette thèse sont évaluées sur la base de vidéos décrite au paragraphe 6.2.2 pour :

– détecter les actions *courir*, *sauter*, *chuter* et *se relever*,
– reconnaître les activités *saut en hauteur*, *saut à la perche*, *saut en longueur* et *triple saut*.

A titre de comparaison, nous appliquons le HMM probabiliste pour les mêmes tâches.

D'une manière générale, quelle que soit la méthode adoptée, la qualité des résultats est particulièrement dépendante de la modélisation choisie pour la conversion numérique-symbolique (i.e. observations-états). Nous appliquons les modélisations décrites dans le paragraphe précédent :

– une approche paramétrique probabiliste (GMM : mélanges gaussiens) dont la sortie est transformée en fonction de croyance si nécessaire,
– une approche crédibiliste basée sur des prototypes (EDC : modèle de distance) dont la sortie est transformée en probabilité si nécessaire,
– une approche crédibiliste experte (EX : règles et événements).

Nous avons appliqué les algorithmes de reconnaissance dans les conditions suivantes (Tab. 6.2) :

**T1 : CSS+EX –** dans cette approche symbolique, on utilise la modélisation experte (EX) pour générer les fonctions de croyance et on reconnaît les séquences avec l'ordonnanceur (CSS).

**T2 : CSS+GMM –** on utilise la modélisation par mélanges Gaussiens (GMM) pour générer les fonctions de croyance (avec application du Théorème de Bayes Généralisé) et on reconnaît les séquences avec l'ordonnanceur (CSS).

**T3 : CrHMM+GMM –** on utilise la modélisation par mélanges Gaussiens (GMM) pour générer les fonctions de croyance (avec application du Théorème de Bayes Généralisé) et on reconnaît les séquences avec le modèle de Markov caché crédibiliste (CrHMM).

**T4 : CrHMM+EDC –** on utilise la modélisation par le modèle de distance crédibiliste (EDC) pour générer les fonctions de croyance et on reconnaît les séquences avec le modèle de Markov caché crédibiliste (CrHMM).

**T5 : HMM+GMM –** on utilise la modélisation par mélanges Gaussiens (GMM) pour générer les vraisemblances et on reconnaît les séquences avec le modèle de Markov caché probabiliste (HMM). Ce test nous sert de référence par rapport à la littérature.

**T6 : HMM+EDC –** on utilise la modélisation par le modèle de distance crédibiliste (EDC) pour générer des fonctions de croyance qui sont ensuite transformées en probabilités pignistiques. On reconnaît les séquences avec le modèle de Markov caché probabiliste (HMM).

Nous reportons les valeurs du rappel (R), de la précision (P) et de la mesure $F_1$ (définis au paragraphe 3.5) de la détection des actions et des activités pour ces différentes combinaisons.

### 6.5.1 Réglages des algorithmes

L'utilisation des algorithmes nécessite quelques réglages décrits ci-après.

**Topologie des séquences** Pour chaque type d'activité, un HMM, un CrHMM et un ordonnanceur CSS sont estimés. Ces modèles de séquences sont composés de $4$ états pour les *sauts en hauteur* (hj : *highjump*), les *sauts à la perche* (pv : *polevault*) et les *sauts en longueur* (lj : *longjump*) : *courir* $\rightarrow$ *sauter* $\rightarrow$ *chuter* $\rightarrow$ *se relever*. Les *triples sauts* (tj : *triple jump*) sont modélisés par des séquences de $8$ états dans le cas du CSS : *courir* $\rightarrow$ *sauter* $\rightarrow$ *chuter* $\rightarrow$ *sauter* $\rightarrow$ *chuter* $\rightarrow$ *sauter* $\rightarrow$ *chuter* $\rightarrow$ *se relever*. Dans le cas du HMM et du CrHMM, le *triple saut* est modélisé par une séquence de $4$ états en supposant que les trois actions *sauter* se ressemblent ainsi que les trois actions *chuter*. Notons que, dans le cadre des HMM et des CrHMM, toutes les transitions possibles entre ces états sont autorisées (modèle ergodique) même si la représentation proposée est séquentielle, l'algorithme de Viterbi permettant de retrouver la séquence optimale.

**Réglages des HMM et des CrHMM** Les paramètres des HMM (matrices de transition et a priori) ainsi que les mélanges Gaussiens utilisés pour lier les caractéristiques aux actions sont estimés à l'aide des algorithmes présentés dans le chapitre 5 et en utilisant $50\%$ de la base pour l'apprentissage.

Les matrices de transition des CrHMM sont construites en partant des matrices probabilistes et en appliquant la méthode de transformation en fonctions de croyance basée sur le PMI décrite dans le chapitre 2-§ 2.1.3. Puis, nous appliquons un affaiblissement généralisé (chap. 2-§ 2.2.6) afin de redistribuer de la masse sur tous les ensembles et sous-ensembles d'états (les coefficients sont égaux à $0.1$).

**Réglages des ordonnanceurs CSS** Afin de faciliter les réglages, nous supposons dans un premier temps que pour un même ordonnanceur (i.e. pour un même type de saut), les paramètres sont identiques. A partir de cette hypothèse, nous pouvons effectuer une étude conjointe simplifiée de l'influence, sur la mesure $F_1$, des paramètres principaux : le seuil de stop de chaque action composant l'ordonnanceur et la pondération de la somme cumulée du conflit associée. L'étude conjointe a été expliquée dans le chapitre 3- § 3.5 pour le cas du filtre. Pour l'étude, nous prenons $50\%$ de l'ensemble d'apprentissage. La figure 6.21 montre que, pour un même réglage pour toutes les actions des *sauts en hauteurs* (seuil de stop égal à $3.4$ et pondération égale à $0.83$), les performances obtenues (en termes de mesure $F_1$) sont satisfaisantes. Une figure similaire est obtenue pour les *sauts à la perche* (Fig. 6.22). Il est ensuite possible de modifier ces seuils manuellement autour de ces valeurs et les paramètres des filtres peuvent être un peu différents.

### 6.5.2 Reconnaissance des actions dans les activités avec la méthode symbolique

L'objectif du test est de mettre en valeur l'apport du séquencement et du filtrage intégré à l'ordonnanceur. Nous appliquons la modélisation experte (EX), basée sur les trapèzes, les règles et les événements, à partir de laquelle on applique l'ordonnanceur (CSS) pour la reconnaissance des actions dans deux types de saut : les *sauts en hauteur* et les *sauts à la perche*.

Les résultats avant et après application de l'ordonnanceur CSS sont donnés dans les tableaux 6.3 et 6.4. L'ordonnanceur CSS permet d'augmenter la précision des détections menant à une améliora-

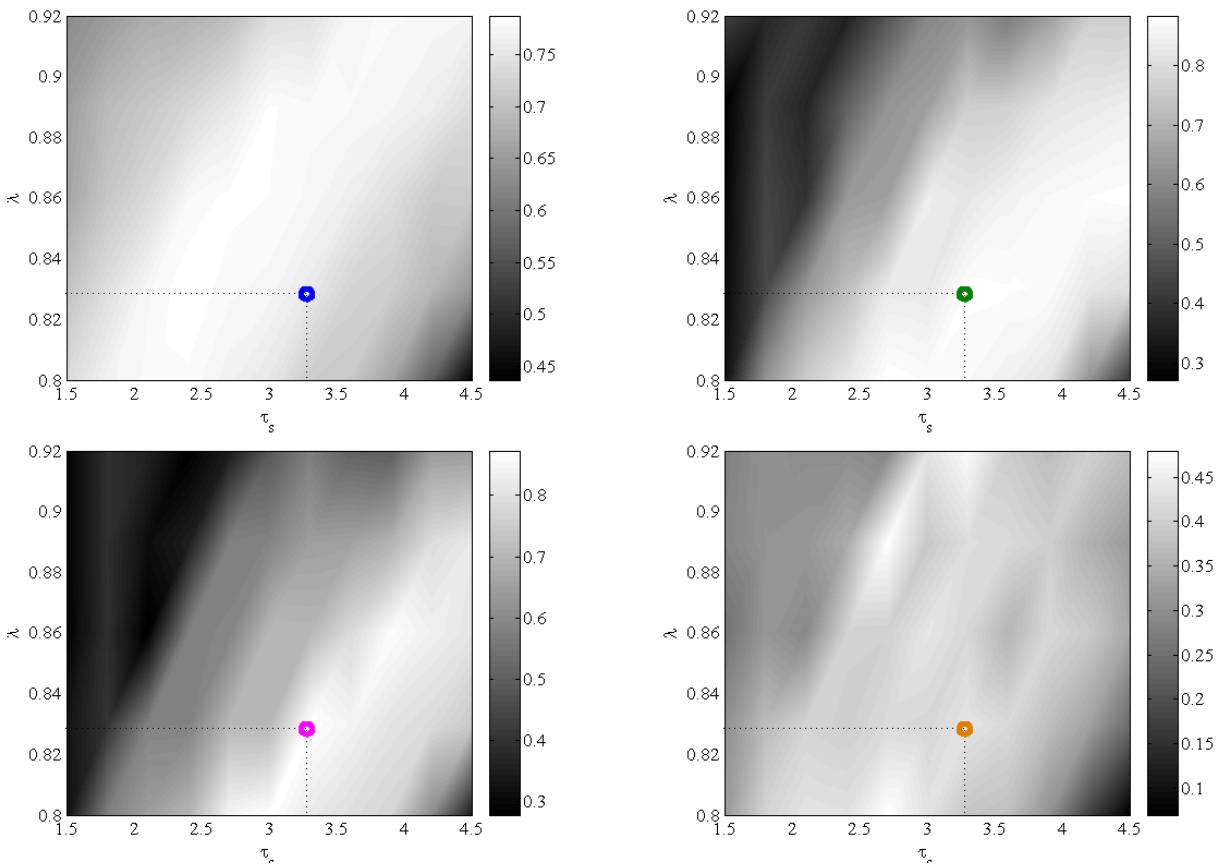



**FIG. 6.21 :** *Evolution de la mesure $F_1$ en fonction du seuil de stop et de la pondération de la somme cumulée du conflit utilisés dans l'ordonnanceur CSS pour des* sauts en hauteur *avec les mélanges gaussiens (GMM) en entrée du système. Ligne 1 : action* courir *et* sauter*, ligne 2 : actions* chuter *et* se relever*. Les points colorés indiquent un même réglage pour les quatre actions dans les* sauts en hauteur *assurant une performance satisfaisante (resp.* $70\%$*,* $80\%$*,* $80\%$ *et* $40\%$ *pour les actions* courir*,* sauter*,* chuter *et* se relever*).*

tion de la détection (mesure $F_1$) de $5\%$ à $32\%$ pour le *saut en hauteur* et jusqu'à $23\%$ pour le *saut à la perche*.

On remarque que le rappel est un peu détérioré. Ceci vient de quelques vidéos où les fausses alarmes sont de taille importante et face auxquelles l'ordonnanceur échoue pour la reconnaissance des actions. Ce type de problème est illustré sur la figure 6.23 où l'action *chuter* est très mal détectée, avec une fausse alarme de taille importante sur l'intervalle $[55, 70]$ qui lance l'ordonnancement dès que l'action *sauter* devient vraie et qui fait donc échouer la détection des actions *sauter*, *chuter* et *se relever*. Dans ce cas, les performances sont nulles. Une solution très simple à ce problème consisterait à dupliquer les séquences au niveau des actions « sensibles » comme les actions *chuter* et *sauter* et donc de différer la décision sur le passage ou non à l'état suivant. Cette solution permettrait d'améliorer grandement les résultats de la détection mais il faudrait gérer les problèmes d'explosion combinatoire si les données sont très bruitées. Ce type de solution s'inspire des méthodes de reconnaissance de chroniques [Vu04, Bre07, DM07].

| | $R$ | $P$ | $F_1$ |
|---|---|---|---|
| courir | 0.94 | 0.59 | 0.73 |
| sauter | 0.95 | 0.50 | 0.66 |
| chuter | 0.59 | 0.32 | 0.41 |
| se relever | 0.48 | 0.20 | 0.28 |

*EX*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.82 | 0.93 | 0.87 |
| 0.86 | 0.60 | 0.71 |
| 0.54 | 0.74 | 0.62 |
| 0.55 | 0.69 | 0.61 |

*CSS+EX*

**TAB. 6.3 :** *Reconnaissance des actions dans les* sauts en hauteur *par le modèle CSS+EX (gain apporté par le CSS pour chaque action par rapport aux observations EX :* $+14\%$, $+5\%$, $+21\%$ *et* $+32\%$*).*

| | $R$ | $P$ | $F_1$ |
|---|---|---|---|
| courir | 0.98 | 0.51 | 0.67 |
| sauter | 0.90 | 0.51 | 0.65 |
| chuter | 0.90 | 0.52 | 0.66 |
| se relever | 0.49 | 0.38 | 0.42 |

*EX*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.85 | 0.97 | 0.90 |
| 0.76 | 0.78 | 0.77 |
| 0.59 | 0.66 | 0.62 |
| 0.89 | 0.48 | 0.62 |

*CSS+EX*

**TAB. 6.4 :** *Reconnaissance des actions dans les* sauts à la perche *par le modèle CSS+EX (gain apporté par le CSS pour chaque action par rapport aux observations EX :* $+23\%$, $+12\%$, $-4\%$ *et* $+20\%$*).*

| | $R$ | $P$ | $F_1$ |
|---|---|---|---|
| courir | 0.72 | 0.90 | 0.80 |
| sauter | 0.64 | 0.81 | 0.71 |
| chuter | 0.79 | 0.46 | 0.58 |
| se relever | 0.77 | 0.44 | 0.56 |

*GMM*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.84 | 0.94 | 0.89 |
| 0.79 | 0.87 | 0.82 |
| 0.86 | 0.62 | 0.72 |
| 0.92 | 0.75 | 0.83 |

*CrHMM+GMM*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.82 | 0.89 | 0.85 |
| 0.75 | 0.84 | 0.79 |
| 0.83 | 0.62 | 0.71 |
| 0.79 | 0.64 | 0.71 |

*HMM+GMM*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.77 | 0.90 | 0.83 |
| 0.69 | 0.81 | 0.75 |
| 0.75 | 0.80 | 0.78 |
| 0.65 | 0.94 | 0.77 |

*CSS+GMM*

**TAB. 6.5 :** *Reconnaissance des actions dans les* sauts en hauteur *avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS.*

| | $R$ | $P$ | $F_1$ |
|---|---|---|---|
| courir | 0.64 | 0.57 | 0.60 |
| sauter | 0.25 | 0.68 | 0.36 |
| chuter | 0.65 | 0.36 | 0.46 |
| se relever | 0.06 | 0.16 | 0.09 |

*EDC*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.62 | 0.58 | 0.60 |
| 0.29 | 0.86 | 0.43 |
| 0.69 | 0.36 | 0.48 |
| 0.13 | 0.29 | 0.18 |

*CrHMM+EDC*

| $R$ | $P$ | $F_1$ |
|---|---|---|
| 0.61 | 0.55 | 0.58 |
| 0.18 | 0.79 | 0.29 |
| 0.71 | 0.36 | 0.48 |
| 0.08 | 0.23 | 0.11 |

*HMM+EDC*

**TAB. 6.6 :** *Reconnaissance des actions dans les* sauts en hauteur *avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités.*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| courir | 0.78 | 0.88 | 0.83 | 0.83 | 0.95 | 0.88 | 0.84 | 0.91 | 0.88 | 0.76 | 0.94 | 0.84 |
| sauter | 0.72 | 0.49 | 0.58 | 0.77 | 0.58 | 0.66 | 0.76 | 0.59 | 0.66 | 0.62 | 0.51 | 0.56 |
| chuter | 0.65 | 0.56 | 0.60 | 0.79 | 0.64 | 0.71 | 0.73 | 0.63 | 0.68 | 0.50 | 0.79 | 0.61 |
| se relever | 0.89 | 0.59 | 0.71 | 0.92 | 0.70 | 0.80 | 0.88 | 0.71 | 0.79 | 0.75 | 0.72 | 0.74 |
| | *GMM* | | | *CrHMM+GMM* | | | *HMM+GMM* | | | *CSS+GMM* | | |

**TAB. 6.7 :** *Reconnaissance des actions dans les* sauts en longueur *avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS.*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|
| courir | 0.63 | 0.58 | 0.60 | 0.65 | 0.61 | 0.63 | 0.66 | 0.60 | 0.63 |
| sauter | 0.19 | 0.19 | 0.19 | 0.16 | 0.15 | 0.15 | 0.15 | 0.16 | 0.15 |
| chuter | 0.37 | 0.30 | 0.33 | 0.39 | 0.31 | 0.34 | 0.39 | 0.31 | 0.35 |
| se relever | 0.34 | 0.40 | 0.37 | 0.38 | 0.46 | 0.42 | 0.39 | 0.45 | 0.42 |
| | *EDC* | | | *CrHMM+EDC* | | | *HMM+EDC* | | |

**TAB. 6.8 :** *Reconnaissance des actions dans les* sauts en longueur *avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités.*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| courir | 0.81 | 0.95 | 0.88 | 0.83 | 0.95 | 0.89 | 0.81 | 0.96 | 0.88 | 0.78 | 0.97 | 0.86 |
| sauter | 0.76 | 0.71 | 0.73 | 0.77 | 0.75 | 0.76 | 0.78 | 0.76 | 0.77 | 0.89 | 0.78 | 0.83 |
| chuter | 0.79 | 0.69 | 0.74 | 0.80 | 0.72 | 0.76 | 0.80 | 0.71 | 0.75 | 0.67 | 0.78 | 0.72 |
| se relever | 0.79 | 0.42 | 0.55 | 0.83 | 0.55 | 0.66 | 0.82 | 0.52 | 0.64 | 0.78 | 0.74 | 0.76 |
| | *GMM* | | | *CrHMM+GMM* | | | *HMM+GMM* | | | *CSS+GMM* | | |

**TAB. 6.9 :** *Reconnaissance des actions dans les* sauts à la perche *avec les mélanges gaussiens (GMM, première série de tableaux) et le modèle de distance (EDC, seconde série de tableaux) en entrée. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par les deux types de modèle. Troisième colonne : application du modèle de Markov caché probabiliste (HMM).*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|
| courir | 0.60 | 0.65 | 0.62 | 0.53 | 0.67 | 0.59 | 0.53 | 0.67 | 0.59 |
| sauter | 0.51 | 0.58 | 0.54 | 0.54 | 0.58 | 0.56 | 0.54 | 0.58 | 0.56 |
| chuter | 0.62 | 0.30 | 0.40 | 0.68 | 0.29 | 0.41 | 0.68 | 0.29 | 0.41 |
| se relever | 0.10 | 0.42 | 0.16 | 0.09 | 0.69 | 0.17 | 0.10 | 0.69 | 0.17 |
| | *EDC* | | | *CrHMM+EDC* | | | *HMM+EDC* | | |

**TAB. 6.10 :** *Reconnaissance des actions dans les* sauts à la perche *avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités.*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| courir | 0.68 | 0.70 | 0.69 | 0.72 | 0.77 | 0.75 | 0.70 | 0.78 | 0.74 | 0.63 | 0.70 | 0.66 |
| sauter | 0.64 | 0.30 | 0.41 | 0.69 | 0.33 | 0.45 | 0.69 | 0.32 | 0.44 | 0.42 | 0.36 | 0.39 |
| chuter | 0.47 | 0.38 | 0.42 | 0.47 | 0.43 | 0.45 | 0.43 | 0.39 | 0.41 | 0.35 | 0.50 | 0.41 |
| se relever | 0.92 | 0.45 | 0.61 | 0.95 | 0.59 | 0.73 | 0.95 | 0.57 | 0.71 | 0.52 | 0.42 | 0.47 |
| | *GMM* | | | *CrHMM+GMM* | | | *HMM+GMM* | | | *CSS+GMM* | | |

**TAB. 6.11 :** *Reconnaissance des actions dans les* triple sauts *avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS.*

| | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ | $R$ | $P$ | $F_1$ |
|---|---|---|---|---|---|---|---|---|---|
| courir | 0.54 | 0.37 | 0.44 | 0.42 | 0.40 | 0.42 | 0.42 | 0.44 | 0.43 |
| sauter | 0.35 | 0.34 | 0.35 | 0.51 | 0.35 | 0.42 | 0.52 | 0.30 | 0.38 |
| chuter | 0.35 | 0.39 | 0.37 | 0.37 | 0.29 | 0.34 | 0.38 | 0.30 | 0.33 |
| se relever | 0.42 | 0.26 | 0.32 | 0.43 | 0.25 | 0.33 | 0.43 | 0.26 | 0.32 |
| | *EDC* | | | *CrHMM+EDC* | | | *HMM+EDC* | | |

**TAB. 6.12 :** *Reconnaissance des actions dans les* triples sauts *avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités.*

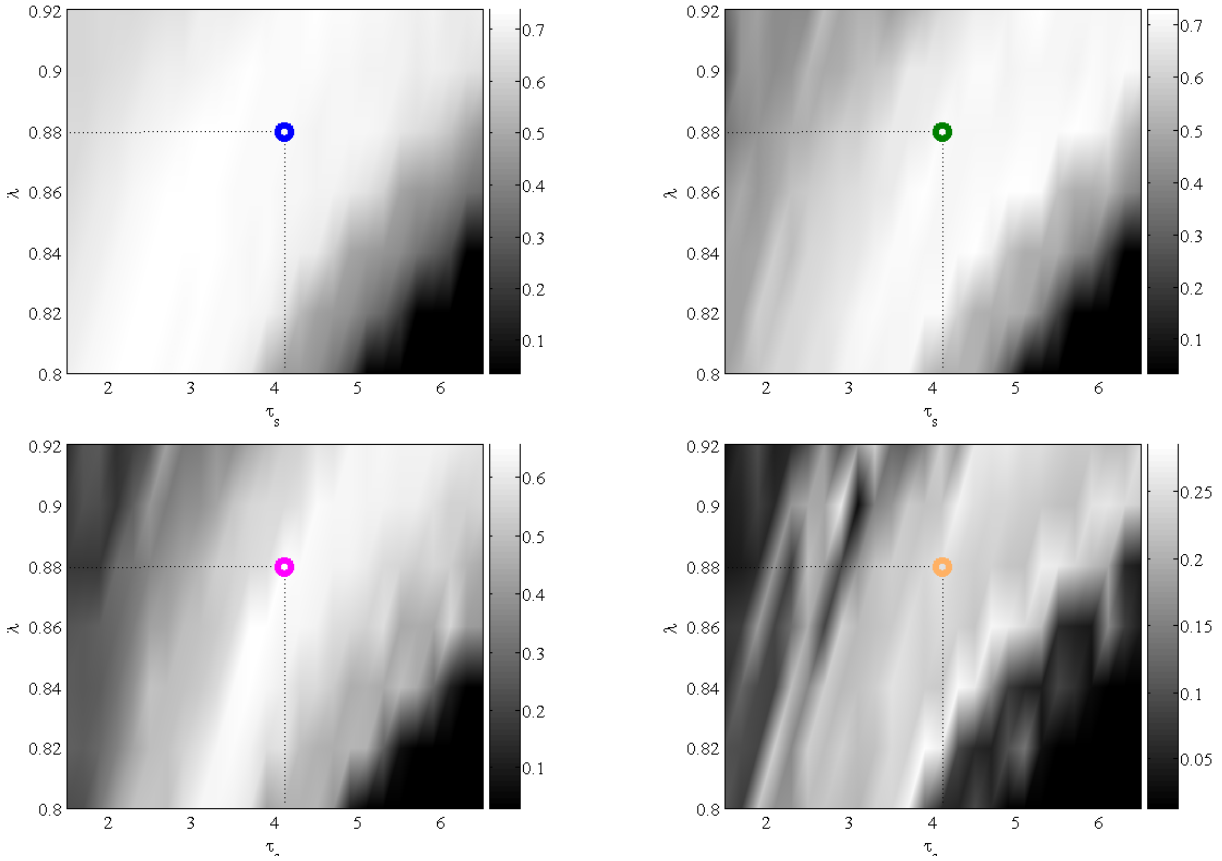



**FIG. 6.22 :** *Evolution de la mesure $F_1$ en fonction du seuil de stop et de la pondération de la somme cumulée du conflit utilisés dans l'ordonnanceur CSS pour des* sauts à la perche *avec les mélanges gaussiens (GMM) en entrée du système. Ligne 1 : action* courir *et* sauter*, ligne 2 : actions* chuter *et* se relever*. Les points colorés indiquent un même réglage pour les quatre actions dans les* sauts à la perche *assurant une performance satisfaisante (resp.* $70\%$*,* $70\%$*,* $60\%$ *et* $25\%$ *pour les actions* courir*,* sauter*,* chuter *et* se relever*).*

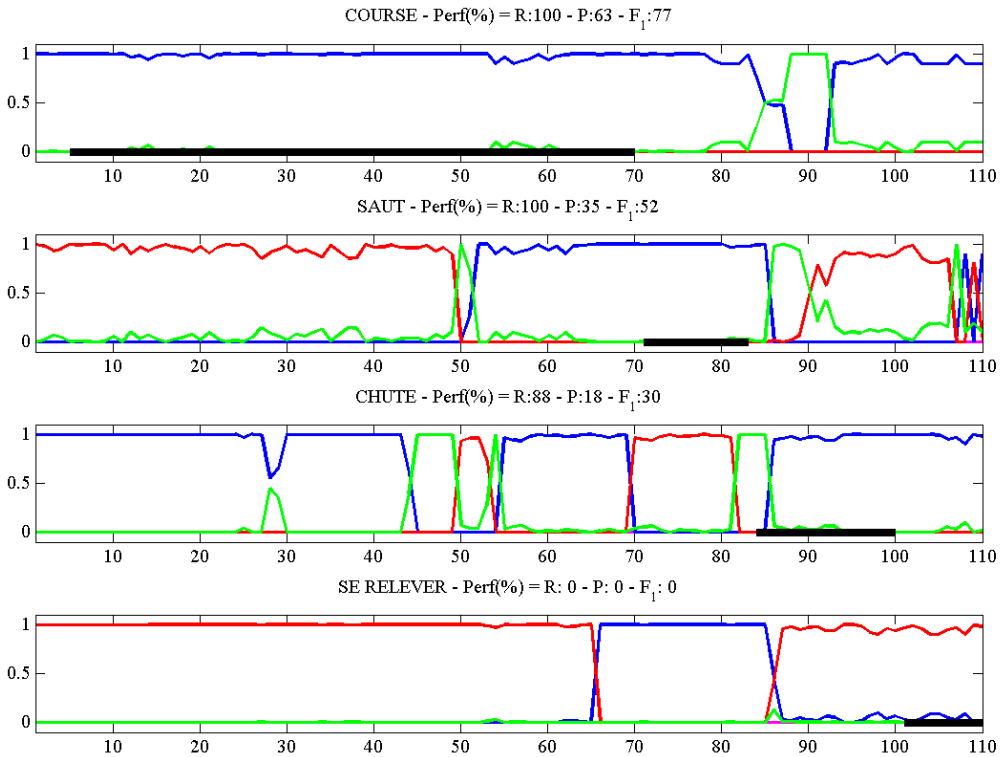
### 6.5.3 Reconnaissance des actions dans les activités avec des méthodes statistiques

Dans ce test, nous étudions les performances du CrHMM et de l'ordonnanceur CSS pour la reconnaissance des actions dans les quatre types de sauts (*sauts en hauteur*, *sauts à la perche*, *sauts en longueur* et *triples sauts*) et les résultats sont comparés à ceux obtenus avec le HMM probabiliste. Nous utilisons les deux modélisations statistiques : les mélanges gaussiens probabilistes (GMM) et modèle de distance crédibiliste (EDC), et nous étudions leur impact sur les résultats.

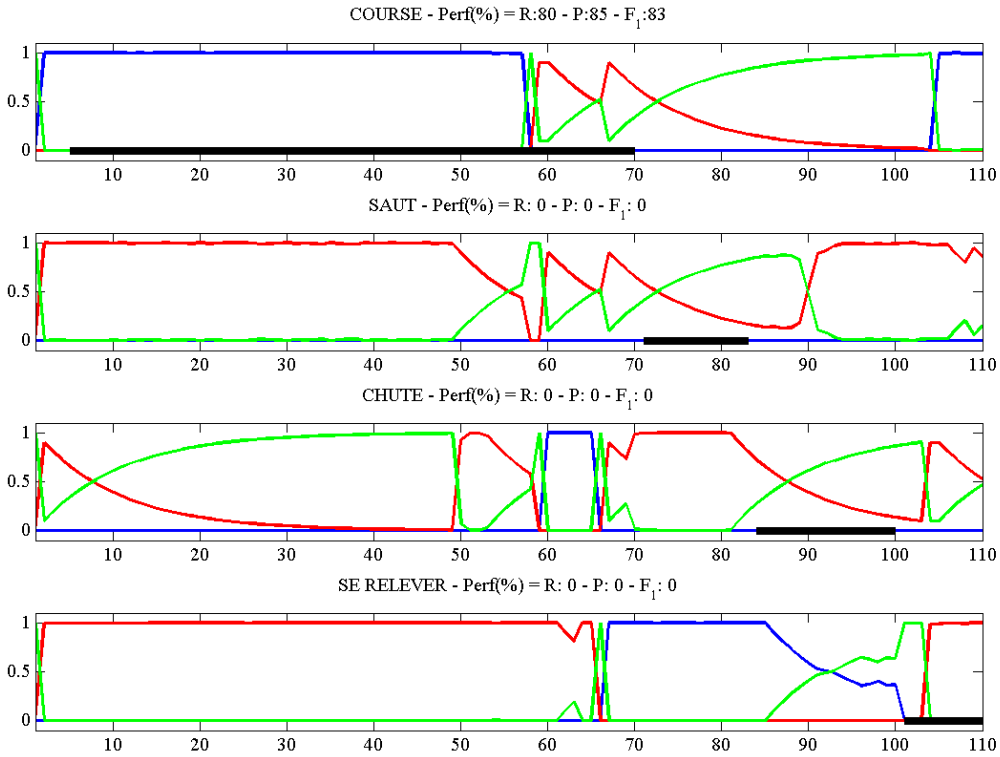
#### 6.5.3.1 Reconnaissance des actions dans les *sauts en hauteur*

Le modèle CrHMM+GMM (Tab. 6.5, col. 2) donne de meilleurs résultats que le modèle HMM+GMM (Tab. 6.5, col. 3) pour les quatre actions et notamment pour l'action *se relever* ($+12\%$). On peut remarquer qu'à la fois le rappel et la précision sont améliorés pour toutes les actions. L'amélioration du CrHMM provient de la propagation dans l'algorithme de Viterbi comme nous l'expliquons dans l'exemple 27.

Le modèle CSS+GMM (Tab. 6.5, col. 4) donne des résultats très satisfaisants, meilleurs que le




*Avant ordonnancement.*




*Après ordonnancement.*

**FIG. 6.23 :** *Erreur de l'ordonnanceur CSS. Une chute est détectée en* $[55, 70]$ *perturbant ainsi le déroulement de la séquence.*

HMM pour les actions *chuter* et *se relever* mais un peu moins bons ($-6\%$) pour *courir* et *sauter*. L'avantage du CSS est de pouvoir contraindre plus aisément la transition entre des états et d'être robuste aux fausses alarmes d'où l'augmentation de l'indice de précision. La diminution de la performance sur les actions *courir* et *sauter* intervient sur le rappel ce qui signifie que la transition *courir* $\rightarrow$ *sauter* et la transition *sauter* $\rightarrow$ *chuter* ont été effectuées trop tard parce que les seuils de stop sont trop grands. Cependant, des seuils plus faibles entrainent une sensibilité plus importante aux fausses alarmes.

Le modèle CrHMM+EDC (Tab. 6.6, col. 2) donne là encore de meilleurs résultats pour les actions *sauter* ($+14\%$) et *se relever* ($+7\%$) que le modèle HMM+EDC (Tab. 6.5, col. 3). Une explication de cet écart, plus important encore que pour le modèle GMM, provient de la transformation croyances-probabilités qui est nécessaire pour appliquer le HMM sur les informations fournies par le modèle EDC. Cette transformation a modifié le contenu d'information puisque la masse de croyance sur le doute est équirépartie sur les singletons. Dans le cas du CrHMM, le doute est conservé et son importance est capitale comme le montre l'exemple 27. Notons que les faibles performances performances de l'EDC sont certainement liées aux données d'apprentissage qui ne sont pas suffisamment importantes ou représentatives. Ses performances pourraient sans doute être améliorées en modifiant les valeurs du nombre de voisins ($K$) ou du coefficient $\alpha$. L'important dans ces tests est la mise en évidence de l'apport du séquencement et du filtrage à partir de modélisations plus ou moins grossières.

**Exemple 27.** *Pour expliquer pourquoi le CrHMM donne généralement de meilleurs résultats que le HMM, prenons l'exemple des figures 6.24-6.27. La figure 6.24 représente les vraisemblances normalisées sur les états (en entrée du HMM et du CrHMM). La figure 6.25 illustre la sortie de l'algorithme de Viterbi probabiliste que l'on compare à la sortie du Viterbi crédibiliste donné à la figure 6.26.*

*On cherche à analyser la différence (de $+13\%$ sur l'action* se relever *en faveur du CrHMM) apparaissant sur l'intervalle d'images* $[73, 85]$ *où le HMM poursuit l'action* chuter *plus longtemps que le CrHMM qui choisit l'action* se relever.

*Pour expliquer cette différence, nous illustrons sur la figure 6.27 l'ensemble d'états utilisé à chaque instant pour le conditionnement avant propagation dans l'algorithme de Viterbi crédibiliste (Eq. 5.57). On rappelle que lorsque cet ensemble est de cardinal élevé, cela signifie que chaque état à un instant $t$ possède un prédécesseur différent. On peut remarquer que sur l'intervalle* $[73, 85]$*, les ensembles utilisés pour le conditionnement contiennent systématiquement l'état* se relever. *Plus précisément, sur* $[79, 85]$*, l'ensemble est $Ch \cup R$ (*chuter *ou* se relever*) puis sur* $[73, 79]$*, l'ensemble est $S \cup Ch \cup R$ (*sauter *ou* chuter *ou* se relever*). Ainsi, l'état* se relever *fait toujours parti des prédécesseurs dans le cas crédibiliste.*

*Dans le cas probabiliste, ce n'est pas le cas à cause de deux raisons :*

1. *la vraisemblance de l'action* chuter *devient supérieure à celle de l'action* se relever *dès l'instant $t = 84$ ($4^e$ courbe de la figure 6.24),*
2. *la probabilité de transition de l'action* chuter *vers l'action* se relever *est plus faible que la probabilité de transition de l'action* chuter *vers elle-même (c'est un fait statistiquement évalué sur l'ensemble d'apprentissage des HMM).*

*Par conséquent, dès que l'action* chuter *devient vraie, il plus difficile pour le HMM d'aller dans l'action* se relever *à moins que sa vraisemblance soit importante, ce qui n'est pas le cas dans cet exemple car sur* $[74, 80]$*, sa vraisemblance reste proche de celle de* chuter *($3^e$ et $4^e$ courbes de la*

*figure 6.24). Dans le cas crédibiliste, l'action* chuter *est vraie mais de la masse de croyance est conservée sur l'union des actions ($Ch \cup R$) ce qui signifie que les deux actions sont encore possibles. L'information portée par le doute permet donc d'être robuste aux fausses alarmes en repoussant la décision.*

*Cette observation permet d'expliquer pourquoi le CrHMM est généralement plus performant que le HMM. Cependant, cela explique aussi pourquoi le CrHMM n'est pas* encore plus *performant : le fait de repousser la décision implique des petits décalages sur les transitions entre états. Ce décalage n'est pas systématique, cela dépend des données d'entrée, par exemple un décalage apparaît au début des actions* sauter *et* se relever *(Fig. 6.25 et 6.26) mais pas au début des actions* courir *et* chuter. *Si les méthodes de modélisation (par exemple GMM et EDC) sont très précises, cela peut entraîner une baisse de performance mais qui reste mineure. Par contre, le CrHMM prend tout son intérêt lors de transitions progressives.*

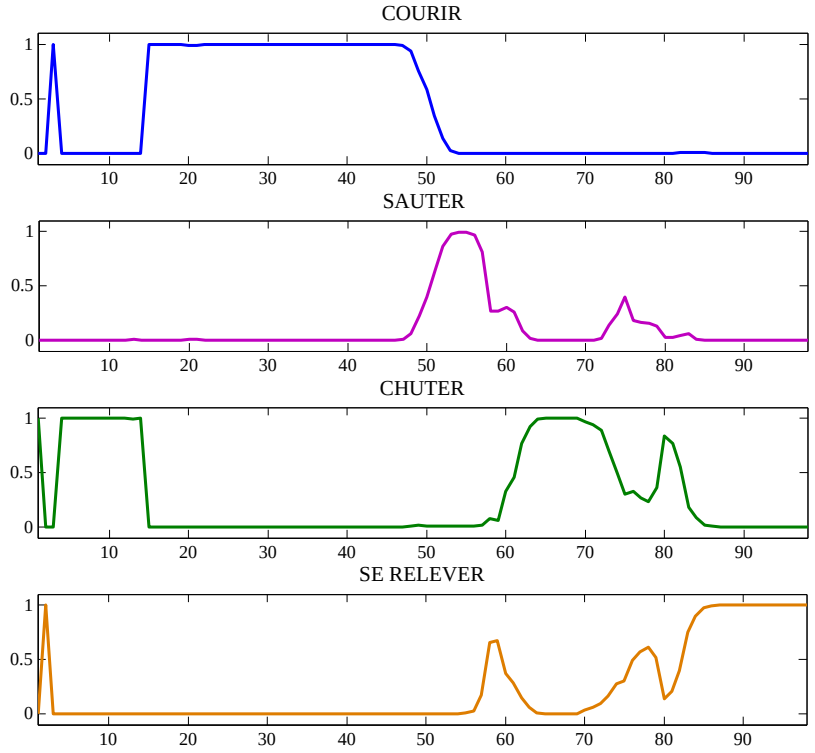



**FIG. 6.24 :** *Les vraisemblances sur les états pour un* saut en hauteur *obtenues à partir des observations.*

#### 6.5.3.2 Reconnaissance des actions dans les *sauts en longueur*

Le modèle CrHMM+GMM (Tab. 6.7, col. 2) est assez proche du modèle HMM+GMM (Tab. 6.7, col. 3) pour cette activité avec une petite amélioration dans le cas de l'action *chuter* ($+3\%$) et *se relever* ($+1\%$). Le modèle CSS+GMM (Tab. 6.7, col. 4) donne des performances correctes mais inférieures aux méthodes stochastiques. Pour le modèle EDC (Tab. 6.8), les résultats sont là encore très proches.

Dans le cas du *saut en longueur*, les croyances en entrée des systèmes sont peu bruitées par conséquent, l'apport du CSS et des autres méthodes pour la détection des actions est peu significatif. Cependant, grâce à ces mécanismes, la séquence des états (optimale par rapport aux paramètres utilisés) est déterminée.

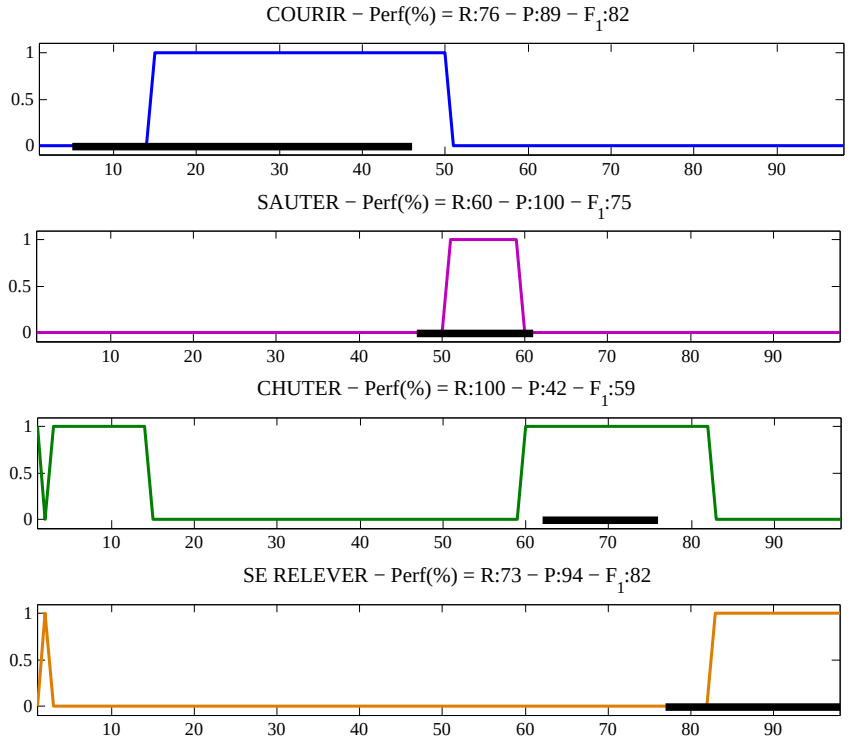



**FIG. 6.25 :** *Décision sur l'état après application de l'algorithme de Viterbi probabiliste.*

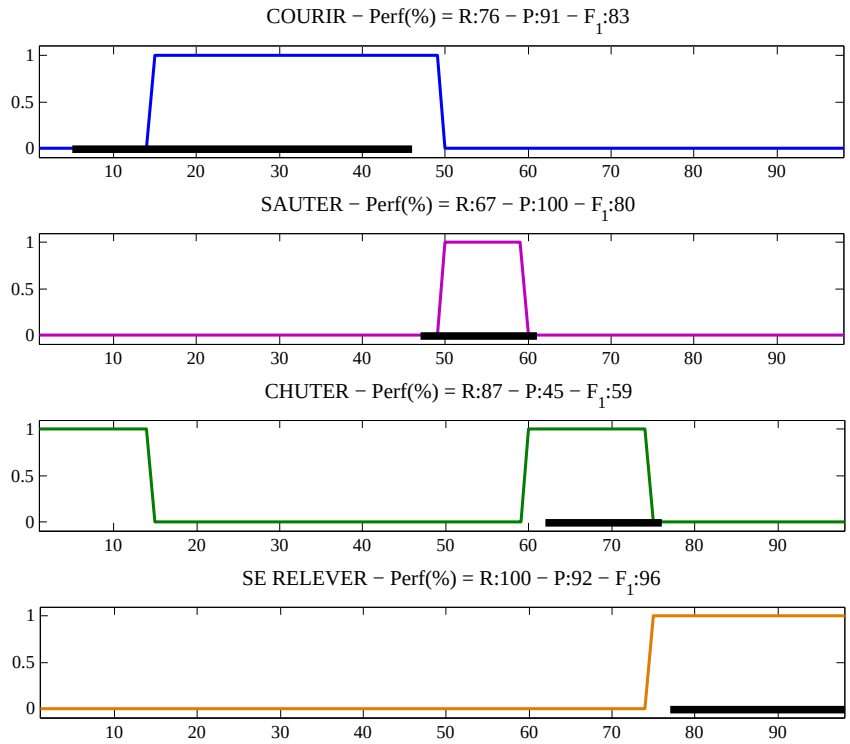



**FIG. 6.26 :** *Décision sur l'état après application de l'algorithme de Viterbi crédibiliste.*

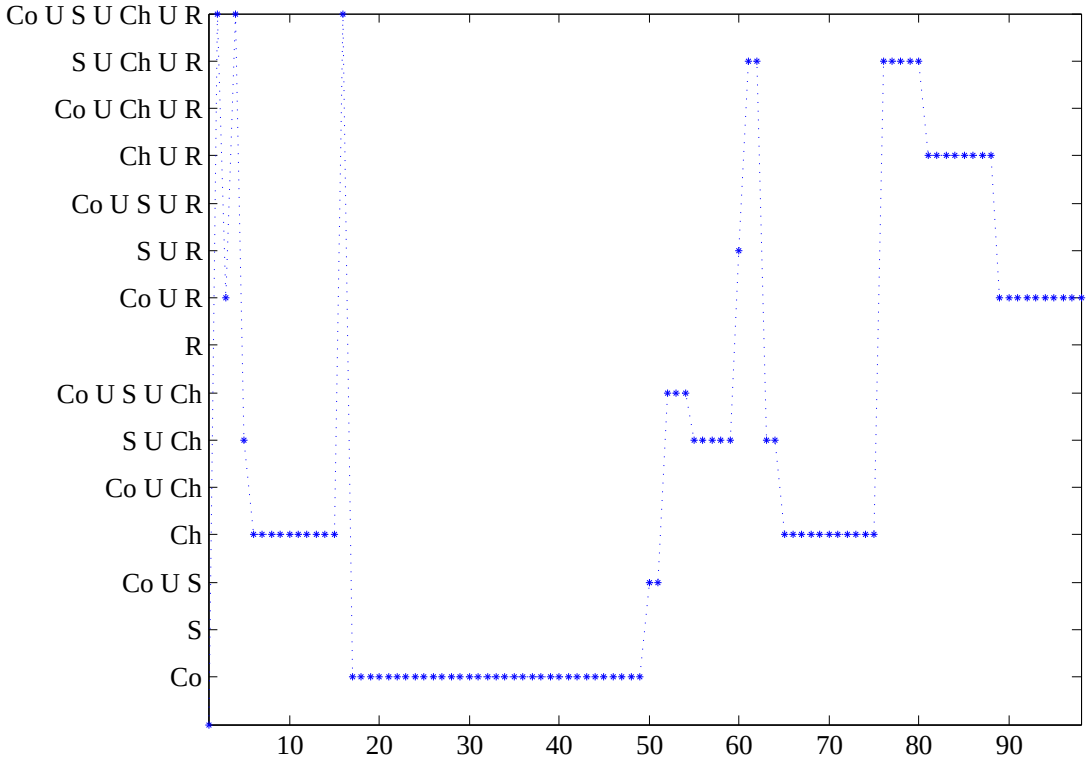



**FIG. 6.27 :** *Ensemble utilisé pour le conditionnement à chaque instant dans l'équation 5.57 où $Co$, $S$, $Ch$ et $R$ signifient respectivement* courir, sauter, chuter *et* se relever. *Les symboles $\cup$ représente une union d'actions.*

#### 6.5.3.3 Reconnaissance des actions dans les *sauts à la perche*

Dans cette activité, les méthodes de détection sont très proches dans le cas de l'EDC (Tab. 6.10). Il en est de même pour la modélisation GMM (Tab. 6.9) avec une petite amélioration dans le cas du modèle CrHMM+GMM (Tab. 6.9, col. 2) : $+3\%$ pour *chuter* et $+1\%$ pour *se relever*, par rapport au modèle HMM+GMM (Tab. 6.9, col. 3). On remarque que le modèle CSS+GMM (Tab. 6.9, col. 4) donne de meilleurs résultats pour les actions *sauter* ($+6\%$) et *se relever* ($+10\%$) tout en restant assez proche mais inférieur aux méthodes stochastiques pour les actions *courir* ($-3\%$) et *chuter* ($-4\%$). La baisse de performance sur ces deux dernières actions est liée au fait que les fausses alarmes sont de taille importante dans le cas des sauts à la perche et par conséquent les seuils ont été réglés à des valeurs élevées entraînant des retards systématiques sur les transitions et donc diminuant le rappel. L'augmentation de la précision vient du fait que les seuils sont justement élevés permettant ainsi de filtrer les fausses alarmes.

#### 6.5.3.4 Reconnaissance des actions dans les *triples sauts*

Dans cette activité, le modèle CrHMM+GMM (Tab. 6.11, col. 2) est très proche ($+1\%$ globalement) du modèle HMM+GMM (Tab. 6.7, col. 3) sauf pour l'action *chuter* où une amélioration de $+4\%$ peut être observée. Dans le cas du modèle CrHMM+EDC (Tab. 6.8, col. 2), les résultats sont là encore très proches ($+1\%$ globalement en faveur du CrHMM) sauf pour l'action *sauter* où une amélioration de $+4\%$ peut être observée. On remarque le modèle CSS+GMM fonctionnne globalement mieux que le meilleur des modèles précédents avec $+23\%$ d'amélioration pour l'action *courir*, $+8\%$ pour l'action *sauter* et $+14\%$ pour l'action *chuter*.

Un gain significatif est obtenu avec le CSS dans le cas du *triple saut* car le CSS est capable

de détecter des actions qui n'ont pas été préalablement détectées en entrée du système. En effet, à partir du moment où une action sur deux est bien détectée, il suffit de régler le paramètre $\Delta_F$ à une valeur élevée sur l'action localisée entre les deux précédentes pour que le CSS comble le trou. Cette caractéristique est à la fois un avantage et un inconvénient car si une fausse alarme apparaît alors le CSS démarre le processus sans pouvoir revenir en arrière.

#### 6.5.3.5 Conclusion sur la détection des actions

Le CrHMM est globalement plus performant que les autres méthodes dans cette application sauf pour quelques cas particuliers où le CSS est un peu meilleur. Le CSS peut être performant si la durée des fausses alarmes apparaissant sur les croyances en entrée du système sont plus faibles que la durée des actions et si la valeur des seuils de stop des filtres utilisés dans l'ordonnanceur est suffisamment élevée. Le problème est que, lorsque l'ordonnancement a débuté, il n'est plus possible de revenir en arrière ce qui peut entraîner parfois des mauvaises détections alors que les méthodes stochastiques comme le CrHMM ne sont pas « piégées » dans la séquence. Ajoutons par ailleurs que le CSS reconnaît les séquences « en ligne », sans rétro-propagation, à la différence de l'algorithme de Viterbi utilisé dans le CrHMM et dans le HMM. Par conséquent, le CSS dispose de moins d'information pour décider de l'état courant puisque la rétropropagation a lieu en fin de séquence mais le CSS dispose cependant d'a priori fort concernant les seuils de transition qui imposent en quelque sorte une durée minimale des actions.

### 6.5.4 Classification de séquences

L'objectif est d'indexer les vidéos, les index étant des séquences d'actions définies par un CSS, un CrHMM ou un HMM.

La classification est réalisée en optimisant un critère propre à l'approche utilisée :

- la maximisation de la vraisemblance de la séquence optimale fournie par l'équation 5.21 dans le cas du HMM,
- la maximisation de la plausibilité de la séquence optimale fournie par l'équation 5.66 dans le cas du CrHMM,
- la maximisation de la qualité globale donnée par l'équation 4.4 dans le cas du CSS.

Pour les trois approches, les observations ont été modélisées par des mélanges gaussiens (GMM) pour chacune des actions puis ont été déclinées sous la forme adaptée à l'approche utilisée : distribution de probabilités pour le HMM et fonctions de croyance pour le CrHMM et le CSS.

Les résultats de la classification sont reportés dans le tableau 6.13. La première remarque concerne l'existence de la classe de rejet pour l'ordonnanceur CSS. Cette classe permet d'éviter de classer des séquences trop « hors profil » alors que les autres approches les classent dans la moins mauvaise classe. Si l'on exclut les cas rejetés, le taux de bonne classification passe alors à $76\%$.

Le *triple saut* est globalement difficile à reconnaître car les actions sont très courtes (environ $5-7$ images) qui peuvent être, dans certains cas, interprétées comme des fausses alarmes.

Intuitivement, on aurait pu s'attendre à ce que les *sauts en hauteur* et les *sauts à la perche* soient confondus, il en est de même pour les *sauts en longueur* et les *triples sauts*. Les résultats montrent que ce n'est pas le cas ce qui signifie que les modèles sont bien discriminants.

Enfin, les meilleures performances de l'ordonnanceur CSS viennent sans doute du critère utilisé

| | Vérité terrain | | | |
|---|---|---|---|---|
| | hj | lj | pv | tj |
| hj | **6** | 5 | 4 | 2 |
| lj | 1 | **14** | 0 | 1 |
| pv | 1 | 1 | **20** | 4 |
| tj | 7 | 1 | 2 | **5** |

*Classification par HMM.*

| | Vérité terrain | | | |
|---|---|---|---|---|
| | hj | lj | pv | tj |
| hj | **9** | 5 | 4 | 3 |
| lj | 1 | **13** | 1 | 1 |
| pv | 3 | 2 | **18** | 2 |
| tj | 2 | 1 | 3 | **6** |

*Classification par CrHMM.*

| | Vérité terrain | | | |
|---|---|---|---|---|
| | hj | lj | pv | tj |
| hj | **14** | 3 | 4 | 0 |
| lj | 1 | **12** | 2 | 2 |
| pv | 0 | 2 | **17** | 0 |
| tj | 0 | 0 | 2 | **8** |
| rej | **0** | **4** | **1** | **2** |

*Classification par CSS.*

**TAB. 6.13 :** *Classification par HMM probabiliste (gauche, taux de bonne classification de* $60\%$*), par HMM crédibiliste (centre, taux de bonne classification de* $60\%$*) et par l'ordonnanceur CSS (droite, taux de bonne classification de* $68\%$*) lorsque les entrées sont fournies par des mélanges gaussiens (GMM).*

qui intègre à la fois le nombre d'états (i.e. le nombre d'actions) qui ont été vrais et le conflit sur ces états lorsque ces derniers ne correspondent pas aux données.

La qualité de la classification dépend essentiellement de la qualité des informations provenant des observations. Pour améliorer les résultats, il peut être intéressant d'utiliser d'autres informations comme la présence d'une perche pendant la course ou bien des durées équivalentes dans les actions *sauter* et *chuter* lors d'un *triple saut*. Cependant, l'utilisation de ces caractéristiques ne peut se faire qu'après une première phase de reconnaissance des actions. A titre d'exemple, nous donnons les résultats obtenus par l'ordonnanceur à partir de données modélisées par l'approche symbolique complétées par les deux détecteurs évoqués ci-dessus (Tab. 6.14). Le taux de bonne classification, en excluant les cas rejetés, passe alors à $85\%$.

| | Vérité terrain | | | |
|---|---|---|---|---|
| | hj | lj | pv | tj |
| hj | **9** | 1 | 3 | 0 |
| lj | 0 | **12** | 0 | 0 |
| pv | 4 | 0 | **23** | 0 |
| tj | 0 | 0 | 0 | **8** |
| rej | 2 | 8 | 0 | 4 |

**TAB. 6.14 :** *Classification par CSS+EX.*

## 6.6 Conclusion

La mise en œuvre sur une application réelle des algorithmes développés dans cette thèse a montré la pertinence de nos approches. Nous nous sommes astreints, dans la mesure du possible, à utiliser les algorithmes avec les mêmes données de base sachant qu'une distribution de probabilités n'est pas équivalente à une fonction de croyance en termes de quantité d'information.

Les tests proposés ont permis d'obtenir des résultats de type statistique tels que les taux de rappel et de précision sur la détection des actions ou encore le nombre de séquences bien classées. Ces tests sont intéressants pour valider un outil d'indexation automatique dans des grandes bases de vidéos.

Les classifieurs spécialisés par type d'action que nous avons utilisés ont été très efficaces puisque dédiés. Cependant, leurs capacités de généralisation restent à étudier. Les cas difficiles de classification sont apparus pour des sauts en vue de face ou par l'arrière qui apparaissent assez rarement (pour les ralentis par exemple). La détection des événements rares pourraient rendre le système plus performant en adaptant des modèles existants ou en créant de nouveaux. Une telle approche semi-supervisée pour la classification a été proposée dans le cadre de l'application des HMM à la détection d'événements rares [ZGPB05].

# Conclusion générale et perspectives

L'indexation automatique de vidéos telle que peut le faire un opérateur humain est difficile à envisager à court terme. Par exemple, à l'INA (Institut National de l'Audiovisuel), cette opération d'indexation de tous les programmes télévisuels est réalisée par plusieurs dizaines d'opérateurs. Ceux-ci seraient très intéressés de disposer d'un outil capable de les assister dans cette tâche fastidieuse. Le problème d'indexation de vidéos peut être partiellement résolu en se focalisant sur une application particulière ou sur un type d'application en proposant des algorithmes dédiés. Dans le cadre de nos travaux de recherche, nous nous sommes intéressés à un type d'application où l'index est modélisé par une séquence d'états. Pour illustrer notre approche, nous avons utilisé des vidéos de sports d'athlétisme et nous avons cherché à les indexer par le mouvement des athlètes.

Les index dynamiques sont représentés par des séquences d'états dans le formalisme des fonctions de croyance afin de gérer l'imperfection des données extraites sur les vidéos. Plus précisément, nous avons utilisé le Modèle des Croyances Transférables (TBM) de Smets qui est un modèle puissant pour la représentation, la combinaison et la prise de décision. Le TBM permet de modéliser aussi bien la connaissance disponible que la méconnaissance et le conflit. Dans le formalisme du TBM, on sépare la phase de modélisation, où la connaissance est représentée par des fonctions de croyance, de la phase de décision, souvent basée sur l'utilisation des probabilités pignistiques. Le formalisme du TBM est un cadre formel mathématiquement bien défini et de nombreux travaux sont en cours pour décliner une version crédibiliste des outils probabilistes classiques.

Nous avons développé deux méthodes originales pour reconnaître des séquences d'états dans le cadre du TBM :

- *une méthode symbolique :* l'ordonnanceur CSS (*Credal State Scheduler*) est adaptée lorsque l'on dispose de connaissances expertes sur les séquences. L'ordonnanceur CSS est une machine à états finis déterministe qui reconnaît des séquences en contraignant la véracité des états. Cette méthode est basée sur un filtre temporel crédibiliste (TCF : *Temporal Credal Filter*) qui peut d'ailleurs être utilisé pour filtrer la croyance « temporelle » sur un état particulier lorsque aucun lien de causalité n'existe entre les états. Le filtre et l'ordonnanceur sont basés sur les modèles à sauts et exploite le conflit entre un modèle d'évolution et les données afin de gérer les commutations de modèles.
- *une méthode statistique :* le modèle de Markov caché crédibiliste (CrHMM) s'avére efficace en présence de données bruitées et lorsque les modèles des séquences sont mal connus a priori. Cette méthode repose sur les mécanismes du modèle de Markov caché probabiliste qui est un standard en reconnaissance de séquences. Le modèle de Markov caché crédibiliste est une machine à états finis stochastique gérant les fonctions de croyance sur les observations comme sur les états et dont les mécanismes généralisent en partie ceux utilisés dans le cas probabiliste.

Ces deux méthodes sont des machines à états dont l'évolution repose sur le principe général « prédiction - mise à jour » utilisé en filtrage dans le cadre probabiliste : la connaissance sur les états

à un instant $t$ dépend de celle sur les états à l'instant précédent ainsi que sur un modèle d'évolution de cette connaissance entre deux instants successifs. Dans le cadre des fonctions de croyance, ce principe s'applique de la même manière que dans le cas probabiliste excepté qu'il est possible de gérer d'une part des croyances explicites sur les unions d'états, ce qui s'avère utile pour la modélisation des transitions entre états, et d'autre part le conflit entre les observations et la prédiction ce qui permet de remettre en question le modèle lorsque ce conflit apparaît de manière fréquente. Dans le cas de l'ordonnanceur et du modèle de Markov caché crédibiliste, ce conflit est exploité comme une information pertinente sur la cohérence du modèle avec les mesures et il permet de classer des séquences.

Les travaux futurs envisagés concernent à la fois des aspects théoriques et des aspects pratiques.

**Aspects théoriques**

*Apprentissage dans le cadre des CrHMM :* il est nécessaire de finaliser la généralisation du modèle de Markov caché crédibiliste, en particulier l'apprentissage des modèles d'observation. Dans le cadre des HMM probabilistes, on procède généralement par une estimation itérative en prenant en compte la *vraisemblance* des données d'apprentissage vis à vis du modèle estimé à chaque itération. Un critère permet ensuite de mettre à jour les modèles tant que cette vraisemblance n'a pas atteint un maximum (local ou global). Il est sans doute possible de mimer un tel processus en estimant par exemple le degré de conflit global entre les données d'apprentissage et le modèle, donné par l'algorithme *forward*, ou encore en calculant la plausibilité de l'ensemble d'apprentissage conditionnellement au modèle, par exemple en utilisant le critère de l'algorithme de Viterbi crédibiliste proposé dans cette thèse.

*Adaptation des modèles :* la topologie des séquences (le nombre d'états, les liens de causalité entre les états ou encore les modèles d'observation) devrait pouvoir évoluer en fonction des observations provenant du système dynamique étudié. L'objectif est de réduire l'annotation des données d'apprentissage et de rendre moins fastidieuse l'étape de modélisation. Une solution consiste à partir de données connues (dont on dispose généralement), puis d'étiqueter les nouvelles données au fur et à mesure de l'évolution du système : on parle d'apprentissage *semi-supervisé*.

**Aspects pratiques, liés à l'indexation et aux interfaces Homme-Machine**

*Enrichissement de la base d'apprentissage :* la base de vidéos que nous avons utilisée était composée de 75 vidéos d'athlétisme ce qui est peu par rapport aux bases de vidéos industrielles telles que celles dont dispose l'INA par exemple. De plus, nous n'avons exploité que les informations visuelles or, les informations textuelles et audio peuvent être pertinentes pour la détection d'événements comme nous l'avons vu dans l'étude bibliographique. Par ailleurs, le TBM n'a pas été encore appliqué pour la fusion multimodale où les problèmes de synchronisation (comme entre l'audio et la vidéo) apparaissent. Enfin, nous pouvons envisager d'appliquer les algorithmes de reconnaissance de séquences à d'autres problèmes que l'analyse de mouvements humains.

*Choix des paramètres :* la reconnaissance nécessite le choix de caractéristiques judicieuses. Les caractéristiques que nous avons choisies ont montré une très bonne pertinence vis à vis des mouvements recherchés mais il pourrait être intéressant de déterminer quelles caractéristiques combiner à chaque instant (au lieu d'appliquer toujours la même combinaison avec les mêmes caractéristiques). Cela pourrait permettre d'ajouter aisément de nouvelles caractéristiques sans pour autant alourdir le processus de reconnaissance. Une solution pourrait consister en la quantification de la quantité d'information apportée par les différentes caractéristiques et en la détermination de l'ensemble de

caractéristiques le plus informatif. Ce type d'approche a été utilisée à quelques reprises dans le cadre de la théorie des probabilités et porte le nom de *fusion active*. Une approche similaire pourrait être employée dans le cadre du Modèle des Croyances Transférables afin d'introduire le doute et le conflit sur les observations.

*Reconnaissance au fil de l'eau :* les algorithmes développés dans cette thèse sont markoviens et peuvent fonctionner « en ligne ». Il pourrait être intéressant de traiter le cas de l'analyse en ligne de vidéos réelles où les informations intéressantes, comme les sauts d'athlétismes, seraient noyées dans les films, les publicités, les journaux télévisés... L'algorithme d'estimation du mouvement de caméra permet déjà de segmenter la vidéo en portions dynamiques. On peut ensuite imaginer appliquer, sur ces portions dynamiques, l'algorithme d'estimation du nombre de personnes dans les vidéos (développé avec l'Université de Crète) pour ne conserver que les portions où un seul athlète mobile évolue. Puis, ces segments seraient analysés par les méthodes de reconnaissance de séquences développés dans cette thèse.

# Annexes

## Listes des annexes

- Annexe 1 : Application pratique de l'extension vide pour la fusion de BBAs avec cadres de discernement hiérarchisés.
- Annexe 2 : Mise à jour une BBA après l'ajout d'une hypothèse dans le cadre de discernement initial.
- Annexe 3 : Exemple d'application du réseau de Pétri crédibiliste et nous montrons l'influence des fausses alarmes sur les transitions.
- Annexe 4 : Interfaces pour l'estimation du mouvement de caméra et pour la reconnaissance des séquences.
- Annexe 5 : Liens vers les logiciels utilisés pendant la thèse.

# Annexe 1 – Fusion de BBAs définies sur des cadres de discernement hiérarchisés

Comme cela a été présenté au chapitre 2, l'extension vide permet de redéfinir une BBA sur un nouvel espace de discernement. Cette opération est particulièrement utile lorsque les espaces sont hiérarchisés.

Prenons l'exemple des instruments de musique qui peuvent être catégorisés de la manière suivante (version simplifiée) : instruments à vent ($V$), à cordes ($C$) et les autres ($H^*$). Chacune de ces catégories contient des sous-catégories. Par exemple, les instruments à vent peuvent être de trois types : à bec sifflet ($B$), à anche ($A$) et à embouchure ($E$). On distingue aussi les anches simples ($A_s$), les anches doubles ($A_d$) et les anches libres ($A_l$). Parmi les instruments à embouchure, on peut avoir des cuivres ($E_c$) et d'autres instruments ($E^*$). Les instruments à cordes peuvent être à cordes pincées ($P$), frottées ($F_t$) ou frappées ($F_p$). Parmi les instruments à cordes frappées, on peut distinguer les instruments à clavier ($F_{pc}$) et sans clavier ($F_{p\bar{c}}$). Cette catégorisation est représentée sous la forme d'un arbre sur la figure 1 avec 3 niveaux de hiérarchie.

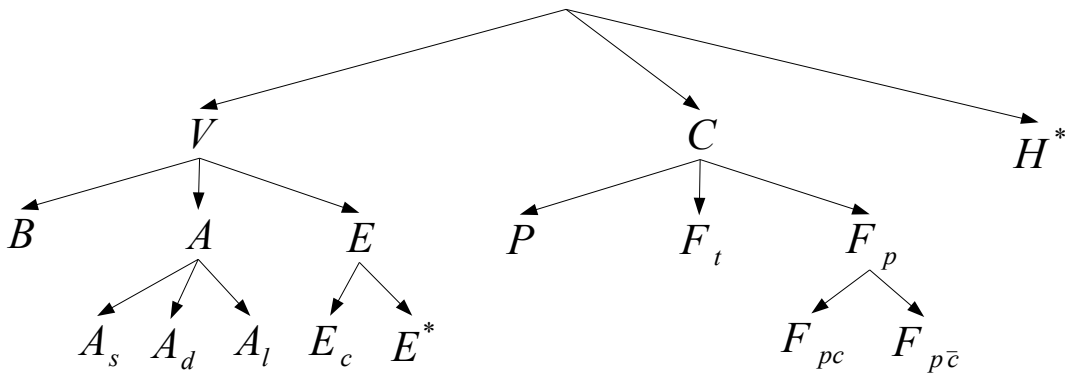



**FIG. 1 :** *Cadres de discernement formant une hiérarchie. Exemple de la catégorisation d'instruments de musique.*

Imaginons que l'on dispose d'analyseurs de sons capables de générer une fonction de croyances sur le type d'instruments à un même niveau de hiérarchie. Suite à l'essai d'un instrument, les analyseurs fournissent trois BBAs : $m_1^{\Omega_1}$, $m_2^{\Omega_2}$ et $m_3^{\Omega_3}$ avec $\Omega_1 = \{V, C, H^*\}$, $\Omega_2 = \{B, A, E, P, F_t, F_p\}$, $\Omega_3 = \{A_s, A_d, A_l, E_c, E^*, F_{pc}, F_{p\bar{c}}\}$. Comment combiner ces BBAs ? Il suffit d'observer que $\Omega_3$ est un raffinement de $\Omega_2$ lui-même étant un raffinement de $\Omega_1$. Par conséquent, la combinaison finale consiste en une succession de combinaisons de BBAs étendues progressivement vers le cadre de discernement le plus raffiné. On commence donc par étendre $m_1^{\Omega_1}$ sur $\Omega_2$ que l'on combine avec $m_2^{\Omega_2}$ puis la BBA obtenue est étendue sur $\Omega_3$ avant d'être combinée avec $m_3^{\Omega_3}$ :

$$m_{123}^{\Omega_3} = m_3^{\Omega_3} \textcircled{\cap} \left( m_2^{\Omega_2} \textcircled{\cap} m_1^{\Omega_1 \uparrow \Omega_2} \right)^{\uparrow \Omega_3} \tag{9}$$

# Annexe 2 – Ajouter des hypothèses dans un cadre de discernement

Comme cela a été présenté au chapitre 2, l'opérateur de déconditionnement peut être utilisé comme un opérateur de changement de cadre de discernement.

Dans l'exemple donné à l'annexe 1 concernant les instruments de musique, on a omis la famille des instruments à percussions ($R$) qui se trouve au même niveau hiérarchique que les instruments à vent et ceux à cordes. Ainsi, le cadre $\Omega$ initialement défini par $\Omega_1 = \{V, C, H^*\}$ devient $\Omega_2 = \{V, C, H^*, R\}$. Imaginons la BBA définie sur $\Omega_1$ par $m^{\Omega_1}$. Elle peut être redéfinie sur $\Omega_2$ de la façon suivante :

$$\begin{array}{rcl} m^{\Omega_1}(V) & = & 0.2 \\ m^{\Omega_1}(\{V,C\}) & = & 0.5 \\ m^{\Omega_1}(\{V,C,H^*\}) & = & 0.3 \end{array} \quad \rightarrow \quad \begin{array}{rcl} m^{\Omega_2}(\{V,R\}) & = & 0.2 \\ m^{\Omega_2}(\{V,C,R\}) & = & 0.5 \\ m^{\Omega_2}(\{V,C,H^*,R\}) & = & 0.3 \end{array} \tag{10}$$

On vérifiera que le conditionnement de la BBA $m^{\Omega_2}$ par $\Omega_1 = \{V, C, H^*\}$ redonne bien la BBA $m^{\Omega_1}$.

Imaginons à présent que l'on dispose d'analyseurs de sons capables de générer une fonction de croyance sur le type d'instruments à différents nœuds de la hiérarchie, i.e. les cadres de discernement sont $\Omega_1 = \{V, C, H^*\}$, $\Omega_{\text{V}} = \{B, A, E\}$, $\Omega_{\text{A}} = \{A_s, A_d, A_l\}$, $\Omega_{\text{E}} = \{E_c, E^*\}$, $\Omega_{\text{C}} = \{P, F_t, F_p\}$ et $\Omega_{\text{F}_p} = \{F_{pc}, F_{p\bar{c}}\}$. Suite à l'essai d'un instrument, les analyseurs fournissent trois BBAs : $m_1^{\Omega_{\text{V}}}$, $m_2^{\Omega_{\text{A}}}$ et $m_3^{\Omega_{\text{E}}}$. Comment combiner ces BBAs ?

On observe que les cadres de discernement $\Omega_{\text{A}}$ et $\Omega_{\text{E}}$ ne partagent pas de raffinement commun : on dit qu'ils sont *incompatibles* [Sha76, DS04]. Le cadre de discernement commun le plus large que l'on puisse trouver est composé de l'*union des cadres de discernement incompatibles* (du niveau le plus bas de la hiérarchie), i.e. $\Omega_3 = \Omega_{\text{A}} \cup \Omega_{\text{E}} = \{A_s, A_d, A_l, E_c, E^*\}$. Ainsi, $\Omega_{\text{A}}$ et $\Omega_{\text{E}}$ sont des *sous-ensembles* du cadre commun $\Omega_3$ qui représente par ailleurs un raffinement de $\Omega_{\text{V}}$. Le problème est à présent de réallouer les masses de croyance sur ces nouveaux cadres. Pour le raffinement, la solution consiste à utiliser l'extension vide. Quant aux cadres incompatibles, il suffit d'observer qu'ils forment des sous-ensembles d'un même cadre et par conséquent, il est possible d'appliquer le déconditionnement comme dans le cas d'un ajout d'hypothèses.

Ainsi, les BBA $m^{\Omega_{\text{A}}}$ et $m^{\Omega_{\text{E}}}$ doivent être *déconditionnées* sur le cadre commun $\Omega_3$ puis combinées conjonctivement. Ensuite, le résultat est combiné avec l'extension vide de la BBA $m^{\Omega_{\text{V}}}$. On obtient donc :

$$m_{123}^{\Omega_3} = \left( m_2^{\Omega_{\text{A}} \Uparrow \Omega_3} \textcircled{\cap} m_3^{\Omega_{\text{E}} \Uparrow \Omega_3} \right) \textcircled{\cap} m_1^{\Omega_{\text{V}} \uparrow \Omega_3} \tag{11}$$

Un problème similaire mais concernant la classification d'automobiles est traité dans [DS04].

# Annexe 3 – Illustration du réseau de Pétri crédibiliste

Le réseau de Pétri crédibiliste (BPN) a été développé pour reconnaître des séquences d'états [RJD99]. Cependant, il est peu robuste aux fausses alarmes comme le montre l'exemple ci-dessous.

On considère la reconnaissance d'une séquence « courir → sauter → chuter → se relever ». Le passage d'un état à un autre se fait par validation d'une transition décrite par une fonction de croyance définie sur un espace binaire (transition vraie ou fausse). La figure 2 illustre les BBA concernant les transitions, la quatrième transition étant toujours vraie (couleur bleue) car elle représente la transition de sortie. La figure 3 illustre les probabilités pignistiques sur les quatre états, chacun des états représentant une action.

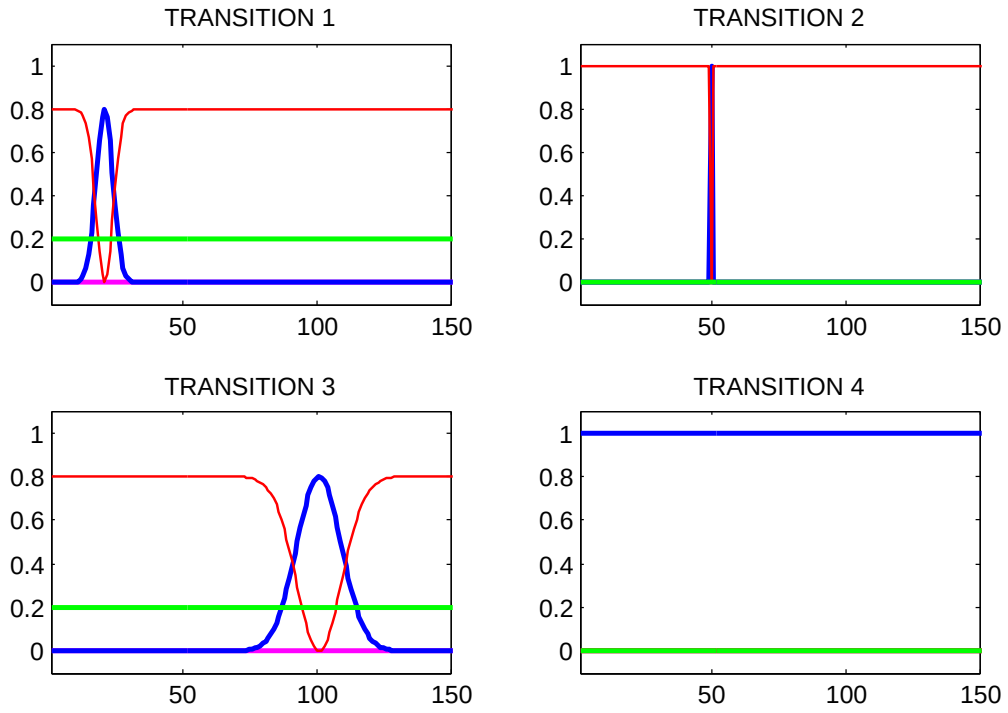



**FIG. 2 :** *Les transitions utilisées dans le réseau de Pétri crédibiliste.*

Le réseau est initialisé avec une croyance vide, i.e. $m^{\Omega}(\{P_1, P_2, P_3, P_4\}) = 1$, c'est pourquoi la pignistique sur tous les états est égale à $0.25$ à $t = 1$. A $t = 20$, la première transition devient vraie progressivement et donc l'action *course* devient fausse tandis que l'action *saut* devient vraie. Lorsque la première transition redevient fausse, la totalité de la croyance du réseau est sur le sous-ensemble $\{P_2, P_3, P_4\}$. Puis, nous avons simulé une fausse alarme sur la deuxième transition à $t = 50$ (Fig. 2 en haut à droite). Cette fausse alarme a pour effet de transférer la masse du réseau sur $\{P_3, P_4\}$ : on voit nettement que la pignistique sur l'action *saut* ($P_2$) décroit instantanément à $t = 50$ (Fig. 3 en haut à droite) tandis que la pignistique sur l'action *chute* augmente instantanément (Fig. 3 en bas à gauche). Puis la troisième transition devient vraie et donc la masse « glisse » vers $P_4$. Dès que la totalité de la masse est dans la dernière place, le réseau n'évoluera plus.

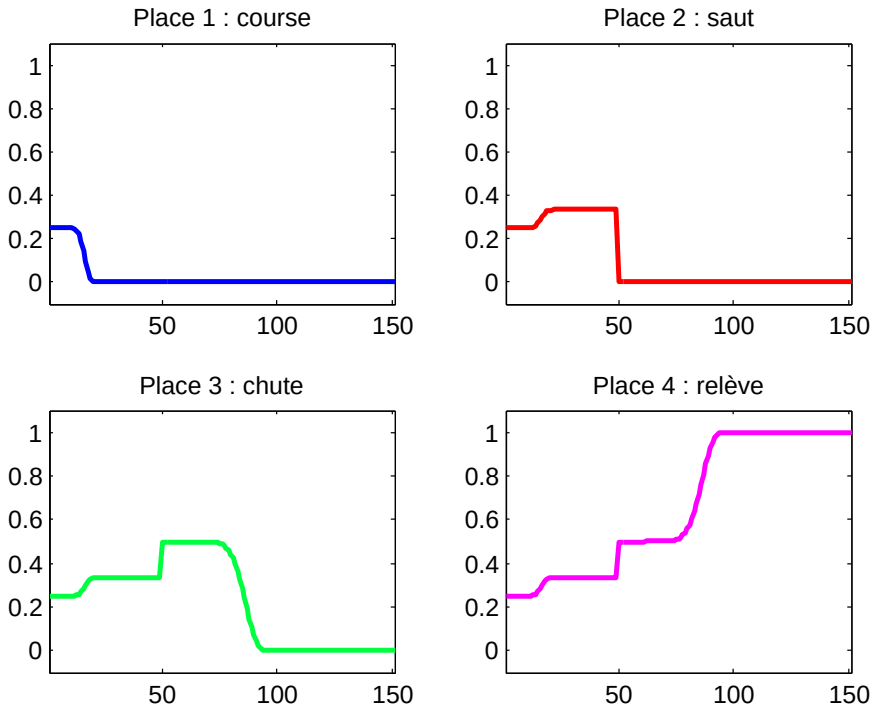



**FIG. 3 :** *Probabilités pignistiques sur les places du réseau de Pétri crédibiliste.*

# Annexe 4 – Interfaces graphiques

L'utilisation de l'estimateur de mouvement de caméra nécessite le réglage de plusieurs paramètres. De plus, le programme *motion2D* développé à l'IRISA offre des fonctionnalités intéressantes comme le masquage de certaines zones de l'image, la compensation de mouvement... Afin de faciliter les réglages pour les tests, nous avons fait développer une interface graphique sous Matlab par un stagiaire Master 1. Cette interface est représentée sur la figure 4.

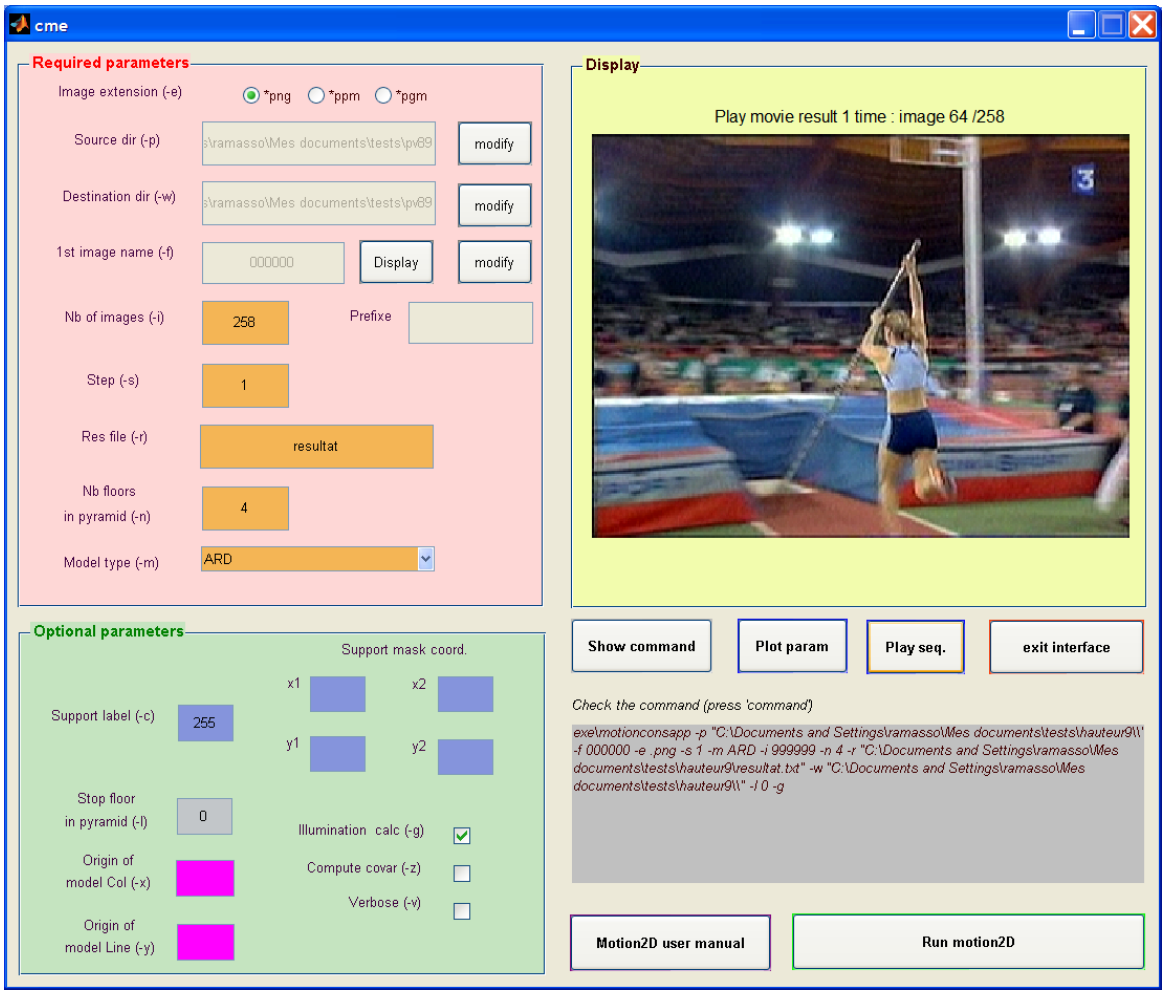



**FIG. 4 :** *Interface graphique pour l'estimation du mouvement de caméra.*

La première fenêtre *Required parameters* permet de rentrer les paramètres indispensables comme le chemin vers le dossier d'images, le type de modèle utilisé... La deuxième fenêtre *Display* permet d'afficher la vidéo originale ou le résultat de l'estimation. La partie *Optional parameters* permet de régler certains paramètres optionnels comme les coordonnées d'un masque à appliquer sur les images. Enfin, nous faisons afficher la commande qui sera exécutée en bas à droite.

Nous avons par ailleurs développé sous Matlab une interface de visualisation des résultats de la reconnaissance de séquences. Cette interface est représentée sur la figure 5. Elle est composée de trois fenêtres : en haut à gauche, nous visualisons l'état courant dans chaque séquence, en haut à droite, la vidéo est affichée et enfin, en bas, un historique des valeurs du critère de reconnaissance pour chaque

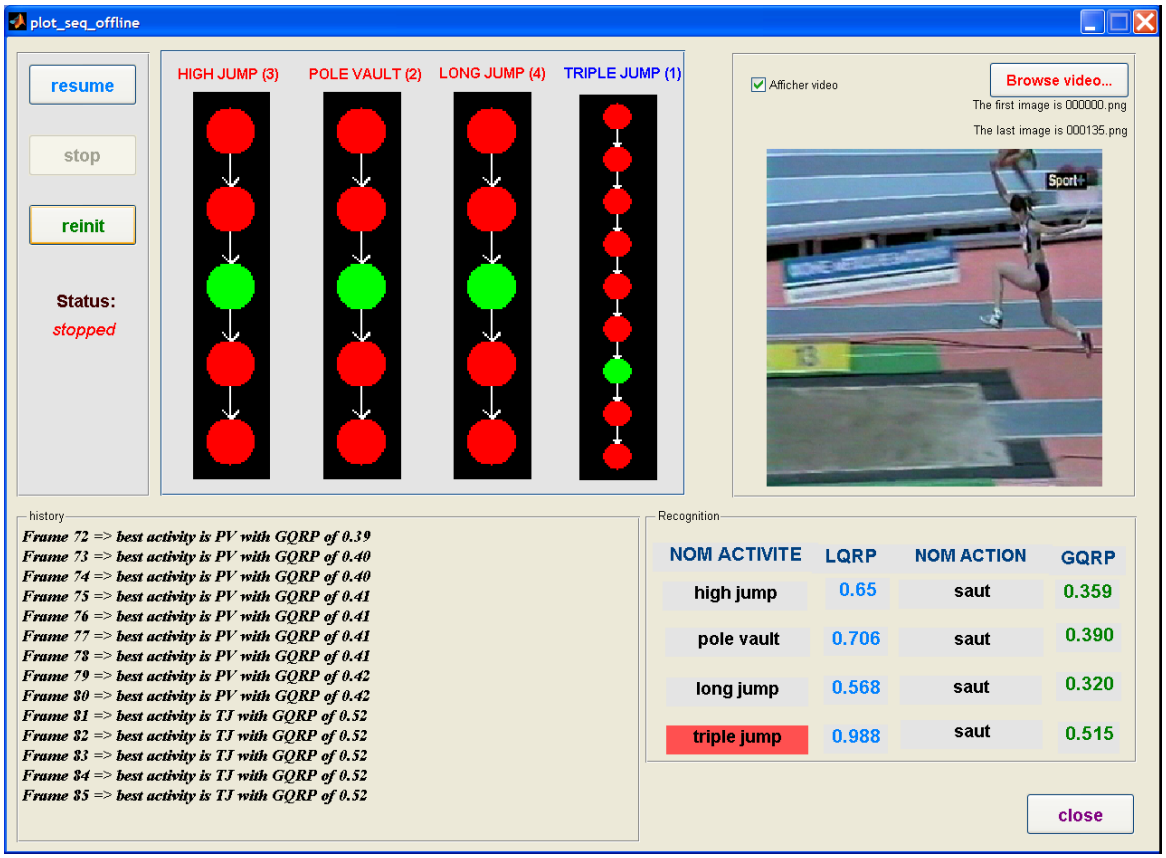



**FIG. 5 :** *Interface graphique pour la reconnaissance des séquences d'états.*

action et pour chaque activité est affiché. Dans l'exemple représenté, l'athlète exécute un *triple saut* et trouve dans la dernière action *sauter*.

# Annexe 5 – Logiciels utilisés

Cette annexe donne les liens vers quelques logiciels utilisés dans cette thèse :

**Motion2D** permet d'estimer le mouvement de caméra dans des séquences d'images [OB95]. Il a été développé en C/C++ par l'INRIA-Rennes. Disponible sur `http://www.irisa.fr/vista/Themes/Logiciel/Motion-2D/Motion-2D.html`.

**Weka** est un ensemble d'algorithmes développés en Java par l'Université de Waikato (Nouvelle-Zélande) pour le clustering et la classification de données [WF05]. Il permet par ailleurs de prétraiter les caractéristiques d'entrées. Disponible sur `http://www.cs.waikato.ac.nz/~ml/weka/`.

**Virtual Dub** est un logiciel de traitement de vidéos développé en C/C++. Il offre des traitements de base comme redimensionner ou filtrer des vidéos mais il permet aussi des traitements plus évolués comme la compression ou la décompression. Il permet notamment de sélectionner des segments de vidéos et d'extraire des séquences d'images. Plusieurs versions de ce logiciel existent permettant de lire des fichiers vidéos de type varié. Disponible sur `http://www.virtualdub.org/`.

**EDC** est le classifieur évidentiel de T. Denœux [Den95, DZ01, VD02] fonctionnant sur Matlab et mis à disposition sur sa page personnelle `http://www.hds.utc.fr/~tdenoeux/`.

**FMT** est un package de fonctions Matlab développées par Ph. Smets permettant de convertir des fonctions de croyance. Disponible sur `iridia.ulb.ac.be/~psmets/`.

**BNT** est une boîte à outils développée par K.P. Murphy offrant des algorithmes d'apprentissage et de reconnaissance de modèles de Markov cachés et de réseaux Bayésiens dynamiques. Disponible sur `www.cs.ubc.ca/~murphyk/`.

# Bibliographie


[AAG02] A.Taleb-Ahmed and L. Gautier. On information fusion to improve segmentation of MRI sequences. *Information Fusion*, 3 :103–117, 2002.

[ABC+03] J. Assfalg, M. Bertini, C. Colombo, A. Del Bimbo, and W. Nunziati. Highlight extraction in soccer videos. In *12th Int. Conf. on Image Analysis and Processing*, 2003.

[ADB06] R.V. Andreao, B. Dorizzi, and J. Boudy. ECG signal analysis through hidden Markov models. *IEEE Trans. on Biomedical Engineering*, 53(8) :1541–1548, 2006.

[AF94] J.F. Allen and G. Ferguson. Actions and events in interval temporal logic. Technical report, University of Rochester, 1994.

[All83] J. Allen. Maintaining knowledge about temporal intervals. *Communications of ACM*, 26 :832–843, 1983.

[AP04] J. K. Aggarwal and S. Park. Human motion : Modeling and recognition of actions and interactions. In *3D Data Processing, Visualization, and Transmission*, pages 640–647, 2004.

[App91] A. Appriou. Probabilités et incertitudes en fusion de données multi-senseurs. *Revue Scientifique et Technique de la Défense*, 11 :27,40, 1991.

[AS01] A. Ayoun and Ph. Smets. Data association in multi-target detection using the Transferable Belief Model. *Int. Jour. of Intelligent Systems*, 16 :1167–1182, 2001.

[BB02] H. Bourlard and S. Bengio. Hidden Markov models and other finite state automata for sequence processing. In Michael A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks : The Second Edition*. The MIT Press, 2002.

[BBN04] M. Bertini, A. Del Bimbo, and W. Nunziati. Common visual cues for sports highlights detection. In *ICME*, pages 1399–1402, 2004.

[BCJ05] N. Benjamas, N. Cooharojananone, and C. Jaruskulchai. Flashlight and player detection in fighting sport for video summarization. In *Proc. of ISCIT*, pages 426–429, 2005.

[BD01] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(3) :257–267, 2001.

[BJA07] T. Battikh, I. Jabri, and M. Annabi. Détection automatique des positions et trajectoires des acteurs d'une recontre de football en vue de mesurer les performances physiques et tactiques. *Can. J. Elect. Comput. Eng.*, 32(2) :113–119, 2007.

[BKOK04] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi. Personalized abstraction of broadcasted american football video by highlight selection. *IEEE Trans. on Multimedia*, 6(4) :575–586, 2004.

[Blo05] I. Bloch. Fusion d'informations numériques : panorama méthodologique. In *Journées Nationales de la Recherche en Robotique*, Guidel, Morbihan, Octobre 2005. `http://jnrr05.irisa.fr/`.

[BLZ+07] L. Bai, S. Lao, W. Zhang, G.J.F Jones, and A.F. Smeaton. A semantic event detection approach for soccer video based on perception concepts and finite state machines. In *Int. Workshop on Image Analysis for Multimedia Interactive Services*, 2007.

[BMF04] M. Bicego, V. Murino, and M.A.T. Figueiredo. Similarity-based classification of sequences using hidden Markov models. *Pattern Recognition*, 37 :2281–2291, 2004.

[BO05] M. Barnard and J.-M. Odobez. Sports event recognition using layered HMMs. In *ICME*, 2005.

[Bou04] P. Bouthemy. 2D motion description and contextual motion analysis : Issues and new models. In *Int. Workshop on Spatial Coherence for Visual Motion Analysis*, Prague, Czech Republic, 2004.

[BR04] R. Venkatesh Babu and K.R. Ramakrishnan. Recognition of human actions using motion history information extracted from the compressed video. *Image and Vision Computing*, 22 :597–607, 2004.

[Bre07] F. Bremond. Interprétation de scènes : perception, fusion multi-capteurs, raisonnement spatio-temporel et reconnaissance d'activités. Habilitation à diriger des recherches (english version), Juillet 2007. Unversité de Nice – Sophia Antipolis.

[BRS+07] M.C. Benitez, J. Ramirez, J.C. Segura, J.M. Ibanez, J. Almendros, A. Garcia-Yeguas, and Guillermo Cortes. Continuous HMM-based seismic-event classification at deception island, antarctica. *IEEE Trans. on Geoscience and Remote Sensing*, 45(1) :138–146, 2007.

[CABT04] F. Cupillard, A. Avanzi, F. Bremond, and M. Thonnat. Video understanding for metro surveillance. In *IEEE special session on Intelligent Transportation Systems*, Taiwan, 2004.

[Cat03] M. Cattaneo. Combining belief functions issued from dependent sources. In *Proc. of the Third Int. Symposium on Imprecise Probabilities and Their Applications*, pages 133–147, Lugano, Switzerland, 2003. Carleton Scientific.

[CC99] O. Chomat and J.L. Crowley. Probabilistic recognition of activity using local appearance. In *CVPR*, pages 104–109, Fort Collins, Juin 1999.

[CCFM04] A.S. Capelle, O. Colot, and C. Fernandez-Maloigne. Evidential segmentation scheme of multi-echo mr images for the detection of brain tumors using neighborhood information. *Information Fusion*, 5 :203–216, 2004.

[CD00] M.-O. Cordier and C. Dousson. Alarm driven monitoring based on chronicles. In *Proc. of the 4th Symposium on Fault Detection Supervision and Safety for Technical Processes*, pages 286–291, Budapest, Hungary, 2000.

[CGPV05] R. Cucchiara, C. Grana, A. Prati, and R. Vezzani. Computer vision system for in-house video surveillance. *IEE Proc. Vis. Image Signal Processing – Intelligent distributed surveillance systems*, 152(2) :242–249, 2005.

[CH06] C.-C. Cheng and C.-T. Hsu. Fusion of audio and motion information on HMM-based highlight extraction for baseball games. *IEEE Trans. on Multimedia*, 8(3) :585–599, 2006.

[Cha05] S. Charbonnier. On line extraction of temporal episodes from ICU high-frequency data : A visual support for signal interpretation. *Computer Methods and Programs in Biomedicine*, 78(2) :115–132, 2005.

[CPV04] R. Cucchiara, A. Prati, and R. Vezzani. Real-time motion segmentation from moving cameras. *Real-Time Imaging*, 10 :127–143, 2004.

[CZ07] B. Chen and W. Zhiqiang. A statistical method for analysis of technical data of a badminton match based on 2-D seriate images. *Tsinghua Science and Technology*, 12(5) :594–602, 2007.

[CZKA96] Y.-L. Chang, W. Zeng, I. Kamel, and R. Alonso. Integrated image and speech analysis for content-based video indexing. In *Proc. of Multimedia*, pages 306–313, 1996.

[DDHJ07] Z. Dang, J. Du, Q. Huang, and S. Jiang. Replay detection based on semi-automatic logo template sequence extraction in sports video. In *Int. Conf. On Image Graphics*, pages 839–844, 2007.

[Dem67] A.P. Dempster. Upper and lower probabilities induced by multiple valued mappings. *Annals of Mathematical Statistics*, 38 :325–339, 1967.

[Dem68] A.P. Dempster. A generalization of Bayesian inference. *Journal of the Royal Statistical Society*, 30 :205–247, 1968.

[Den95] T. Denoeux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Trans. on Systems, Man and Cybernetics*, 5 :804–813, 1995.

[Den00] T. Denoeux. An evidence-theoric neural network classifier. *IEEE Trans. on Systems, Man and Cybernetics*, 3 :712–717, 2000.

[Den04] T. Denoeux. Théorie des fonctions de croyance et classification. Séminaire, thème ASTRID, UTC, Juin 2004.

[Den07] T. Denoeux. Conjunctive and disjunctive combination of belief functions induced by non distinct bodies of evidence. *To appear in Artificial Intelligence*, 2007.

[DF07] Y. Ding and G. Fan. Segmental hidden Markov models for view-based sport video analysis. In *Proc. of Int. Workshop on Semantic Learning Applications in Multimedia, in conjunction with CVPR'07*, June 2007.

[DGG93] C. Dousson, P. Gaborit, and M. Ghallab. Situation recognition : Representation and algorithms. In *Proc. of the 13th IJCAI*, volume 1, pages 166–172, Chambery, France, 1993.

[DGH+02] Z. Duric, W.D. Gray, E. Heishman, F. Li, A. Rosenfeld, M.J. Schoelles, C. Schunn, and H. Wechsler. Integrating perceptual and cognitive modeling for adaptive and intelligent human-computer interaction. *Proc. of the IEEE*, 90(7) :1272–1289, 2002.

[DGPS01] D. Dubois, M. Grabisch, H. Prade, and Ph. Smets. Using the transferable belief model and a qualitative possibility theory approach on an illustrative example : the assessment of the value of a candidate. *Int. Jour. of Intelligent Systems*, 16 :1245–1272, 2001.

[DM04] T. Denoeux and M.H. Masson. EVCLUS : evidential clustering of proximity data. *IEEE Trans. Systems, Man and Cybernetics B*, 34(1), February 2004.

[DM07] C. Dousson and P. Le Maigat. Chronicle recognition improvement using temporal focusing and hierarchization. In *IJCAI*, pages 324–329, 2007.

[Dou96] C. Dousson. Alarm driven supervision for telecommunications network : II – on-line chronicle recognition. *Annales des Télécommunications*, 51(9-10) :501–508, 1996.

[Dou02] C. Dousson. Extending and unifying chronicle representation with event counters. In *ECAI*, pages 257–261, Lyon, France, 2002.

[DP82] D. Dubois and H. Prade. On several representations of an uncertainty body of evidence. In M. M. Gupta and E. Sanchez, editors, *Fuzzy Information and Decision Processes*, pages 167–181, North-Holland, New-York, 1982.

[DP86a] D. Dubois and H. Prade. The principle of minimum specificity as a basis for evidential reasoning. In *Uncertainty in knowledge-based systems*, volume 12, pages 75–84. Springer-Verlag, 1986.

[DP86b] D. Dubois and H. Prade. A set-theoretic view of belief functions : Logical operations and approximations by fuzzy sets. *Int. Jour. of General Systems*, 12 :193–226, 1986.

[DRK03] H. Denman, N. Rea, and A. Kokaram. Content-based analysis for video from snooker broadcasts. *Computer Vision and Image Understanding*, 92 :176–195, 2003.

[DS04] F. Delmotte and Ph. Smets. Target identification based on the Transferable Belief Model interpretation of Dempster-Shafer model. *IEEE Trans. on Systems, Man and Cybernetics*, 34(4), July 2004.

[DS06] T. Denoeux and Ph. Smets. Classification using belief functions : The relationship between the case-based and model-based approaches. *To appear in IEEE Trans. on Systems, Man and Cybernetics*, 2006.

[Duc04] F. Duchene. *Fusion de données multicapteurs pour un système de télésurveillance médicale de personnes à domicile*. PhD thesis, Université Joseph Fourier, Grenoble, France, 2004.

[DY02] T. Denoeux and A.B. Yaghlane. Approximating the combination of belief functions using the fast Moebius transform in a coarsened frame. *Int. Jour. of Approximate Reasoning*, 37 :77–101, 2002.

[DZ01] T. Denoeux and L. M. Zouhal. Handling possibilistic labels in pattern classification using evidential reasoning. *Fuzzy Sets and Systems*, 122(3) :47–62, 2001.

[EBMM03] A.A. Efros, A.C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *IEEE Int. Conf. on Computer Vision*, pages 726–733, Nice, France, 2003.

[EMS01] Z. Elouedi, K. Mellouli, and Ph. Smets. Belief decision trees : Theoretical foundations. *Int. Jour. of Approximate Reasoning*, 28 :91–124, 2001.

[FJ02] M.A.T Figueiredo and A.K. Jain. Unsupervised learning of finite mixture models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24, 2002.

[FLB06] P.J. Figueroa, N.J. Leite, and R.M.L. Barros. Background recovering in outdoor image sequences : An example of soccer players segmentation. *Image and Vision Computing*, 24 :363–374, 2006.

[GBC05] V. Girondel, L. Bonnaud, and A. Caplier. A belief theory based static posture recognition system for real-time video surveillance applications. *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 10–15, September 2005.

[GCB06] V. Girondel, A. Caplier, and L. Bonnaud. A human body analysis system. *European Journal on Applied Signal Processing*, 2006, Article ID 61927, 18 pages 2006.

[GCBR05a] V. Girondel, A. Caplier, L. Bonnaud, and M. Rombaut. Belief theory-based classifiers comparison for static human body postures recognition in video. *Int. Jour. of Signal Processing*, 2(1) :29–33, 2005.

[GCBR05b] V. Girondel, A. Caplier, L. Bonnaud, and M. Rombaut. Belief theory-based classifiers comparison for static human body postures recognition in video. *International Journal of Signal Processing*, 2(1) :29–33, March 2005.

[GG04a] R.D. Green and L. Guan. Quantifying and recognizing human movement patterns from monocular video images - part I : A new framework for modeling human motion. *IEEE Trans. on Circuits and Systems for Video Technology*, 14(2), 2004.

[GG04b] R.D. Green and L. Guan. Quantifying and recognizing human movement patterns from monocular video images - part II : Applications to biometrics. *IEEE Trans. on Circuits and Systems for Video Technology*, 14(2) :191–198, 2004.

[Gir06] V. Girondel. *Contibution à l'analyse et à l'interprétation du mouvement humain : application à la reconnaissance de posture*. PhD thesis, Institut National Polytechnique de Grenoble, June 2006.

[Gra99] D.M. Gravila. The visual analysis of human movement : A survey. *Computer Vision and Image Undertanding*, 73(1) :82–98, 1999.

[Gui06] M. Guironnet. *Méthodes de résumé de vidéo à partir d'informations bas niveau, du mouvement de caméra ou de l'attention visuelle*. PhD thesis, Université Joseph Fourier, Grenoble, France, 2006.

[HA96] B. Hamidzadeh and Y. Atif. Dynamic scheduling of real-time aperiodic tasks on multiprocessor architectures. In *Hawaii Int. Conf. on System Science*, volume 1, pages 469–478, 1996.

[Ham05a] Hammal Z., Caplier A. and Rombaut M. Belief theory applied to facial expressions classification. *Proc. 3rd International Conference on Advances in Pattern Recognition (ICAPR)*, Bath, United Kingdom, 2005.

[Ham05b] Hammal Z., Couvreur L., Caplier A. and Rombaut M. Facial expressions recognition based on the belief theory : Comparison with diferent classifiers. *Proc. 13th International Conference on Image Analysis and Processing. (ICIAP)*, Italy, 2005.

[Ham07] Hammal Z., Couvreur L., Caplier A. and Rombaut M. Facial expressions classification : A new approach based on transferable belief model. *International Journal of Approximate Reasoning (Elsevier)*, (to appear) 2007.

[Han03] A. Hanjalic. Generic approach to highlight detection in a sport video. In *Proc. of the IEEE International Conference on Image Processing*, volume 1, pages 1–4, 2003.

[Har99] D. Harmanec. Measures of uncertainty and information. 1999.

[HBN00] S. Hongeng, F. Bremond, and R. Nevatia. Bayesian framework for video surveillance application. In *Int. Conf. on Pattern Recognition*, Barcelona, 2000.

[HC05] W. Huang and B. Chen. Scheduling of batch plants : Constraint-based approach and performance investigation. *Int. Jour. of Production Economics*, 2005.

[HCR05] Z. Hammal, A. Caplier, and M. Rombaut. Belief theory applied to facial expressions classification. In *Int. Conf. on Advances in Pattern Recognition*, Bath, United Kingdom, 2005.

[HHD00] I. Haritaoglu, D. Harwood, and D. Davis. W4 : Real-time surveillance of people and their activities. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8) :809–830, 2000.

[HHH+07] Q. Huang, J. Hu, W. Hu, T. Wang, H. Bai, and Y. Zhang. A reliable logo detection and replay detector for sports video. In *ICME*, pages 1695–1698, 2007.

[HL03] R. Haenni and N. Lehmann. Implementing belief function computations. *Int. Jour. of Intelligent Systems*, 18(1) :31–49, 2003.

[HNB04] S. Hongeng, R. Nevatia, and F. Bremond. Video-based event recognition and probabilistic recognition methods. *Computer Vision and Image Undertanding*, 96 :129–162, 2004.

[Hon02] S. Hongeng. *A Unified Bayesian and Logical Approach for Video-base Event Recognition*. PhD thesis, University of Southern California, 2002.

[HTWM04] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. on Systems, man and cybernetics C*, 34(3), 2004.

[ICLB06] B.E. Ionescu, D. Coquin, P. Lambert, and Vasile Buzuloiu. Fuzzy semantic action and color characterization of animation movies in the video indexing task context. In S. Marchand-Maillet et al., editor, *AMR*, volume LCNS 4398, pages 119–135. Springer-Verlag, 2006.

[JLBA07] T.D. Jones, S.W. Lawson, D. Benyon, and A. Armitage. Comparison of human and machine recognition of everyday human actions. In V.G. Duffy, editor, *Digital Human Modelling, HCII*, volume 4561, pages 120–129, 2007.

[Joh75] G. Johnsson. Visual motion perception. In *Scientific American*, pages 76–88, 1975.

[JR85] B. Juang and L. Rabiner. A probabilistic distance measure for hidden Markov models. *AT&T Technical Journal*, 64 :391–408, 1985.

[JS05] A. Jaimes and N. Sebe. Multimodal human computer interaction : A survey. In *IEEE Int. Workshop on Human Computer Interaction in conjunction with ICCV*, volume 3766, pages 1–15, Beijing, China, 2005.

[Kel07] M. Keller. Machine learning approach to text representation using unlabelled data. PhD thesis 06-76, IDIAP, Ecole Polytechnique Fédérale de Lausanne, March 2007.

[KFL01] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. on Information Theory*, 47(2) :498–519, 2001.

[KGG+03] E. Kijak, G. Gravier, P. Gros, L. Oisel, and F. Bimbot. HMM based structuring of tennis videos using visual and audio cues. *ICME*, 3 :309–312, 2003.

[KKO03] T. Koyama, I. Kitahara, and Y. Ohta. Live 3D video in soccer stadium. In *IEEE & ACM Int. Symp. on Mixed and Aug. Reality*, pages 178–186, Japon, 2003.

[KL98] G.H. Kim and S.G. Lee. Genetic reinforcement learning approach to the heterogeneous machine scheduling problem. *IEEE Trans on Robotics and Automation*, 14(6) :879–894, 1998.

[KM95] J. Kholas and P.A. Monney. A mathematical theory of hints : An approach to the Dempster-Shafer theory of evidence. In *Lecture Notes in Economics and Mathematical Systems*, volume 425. Springer-Verlag, 1995.

[KS06] M.H. Kolekar and S. Sengupta. Event-importance based customized and automatic cricket highlight generation. In *ICME*, pages 1617–1620, 2006.

[Kun04] L.I. Kuncheva. Classifier ensembles for changing environments. In *Int. Workshop on Multiple Classifier Systems*, volume 3077, pages 1–15, 2004.

[KW99] G.J. Klir and M.J. Wierman. *Uncertainty-based information. Elements of generalized information theory, 2nd edition*. Studies in fuzzyness and soft computing. Physica-Verlag, 1999.

[LCL07] C.-C. Lien, C.-L. Chiang, and C.-H. Lee. Scene-based event detection for baseball videos. *Jour. of Visual Communication and Image Representation*, 18 :1–14, 2007.

[LCV02] E. Lefevre, O. Colot, and P. Vannoorenberghe. Belief function combination and conflict management. *Information Fusion*, 3 :149–162, 2002.

[LEPS04] B. Li, J.H. Errico, H. Pan, and I. Sezan. Bridging the semantic gap in sports video retrieval and summarization. *Jour. of Visual Communication and Image Representation*, 15, 2004.

[LHGT04] L. Li, W. Huang, Y.H. Gu, and Q. Tan. Statistical modeling of complex backgrounds for foreground object detection. *IEEE Trans. on Image Processing*, 13(11), 2004.

[LHV03] S. Lefèvre, J. Holler, and N. Vincent. A review of real-time segmentation of uncompressed video sequences for content-based search and retrieval. *Real-Time Imaging*, 9 :73–98, 2003.

[LL06] W.-L. Lu and J.J. Little. Tracking and recognizing actions at distance. In *ECCV'06 Workshop on Computer Vision Based Analysis in Sport Environments (CVBASE '06)*, Austria, May 2006.

[LMB02] G. Lefaix, E. Marchand, and P. Bouthemy. Motion-based obstacle detection and tracking for car driving assistance. *Int. Conf. on Pattern Recognition*, 4 :74–77, 2002.

[LMI06] L.Zehnic-Manor and M. Irani. Statistical analysis of dynamic actions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(9) :1530–1535, 2006.

[LP05a] P. Lanchantin and W. Pieczynski. Evidential Markov chains and trees with applications to non stationary processes. *Traitement du signal*, 22(1) :1–10, 2005.

[LP05b] P. Lanchantin and W. Pieczynski. Unsupervised restoration of hidden nonstationary Markov chains using evidential priors. *IEEE Trans. on Signal Processing*, 53(8) :3091–3098, 2005.

[LP07] I. Laptev and P. Pérez. Retrieving actions in movies. In *ICCV*, 2007.

[LSE02] M. Lew, N. Sebe, and J. Eakins. Challenges in image and video retrieval. *Lecture notes in Computer Science, ICIVR*, 2383 :1–6, 2002.

[LWH03] Y. Luo, T.D. Wu, and J.N. Hwang. Object-based analysis and interpretation of human motion in sports video sequences by dynamic Bayesian networks. *Computer Vision and Image Undertanding*, 92 :196–216, 2003.

[MB03] M. Milisavljevic and I. Bloch. Sensor fusion in anti-personnel mine detection using a two-level belief function model. *IEEE Trans. Systems, Man and Cybernetics C*, 33(2) :269–283, 2003.

[MB07] B. Mesot and D. Barber. Switching linear dynamical systems for noise robust speech recognition. *IEEE Trans. on Audio, Speech and Language Processing*, 15(6) :1850–1858, 2007.

[MCJ+05] K. Messer, W.J. Christmas, E. Jaser, J. Kittler, B. Levienaise-Obadia, and D. Koubaroulis. A unified approach to the generation of semantic cues for sports video annotation. *Signal Processing*, 85 :357–383, 2005.

[Mer06] D. Mercier. *Fusion d'informations pour la reconnaissance automatique d'adresses postales dans le cadre de la théorie des fonctions de croyance*. PhD thesis, Universite de Technologie de Compiègne, December 2006.

[MFS05] C. Micheloni, G.L. Foresti, and L. Snidaro. A network of cooperative cameras for visuel surveillance. *IEE Proc. Vis. Image Signal Processing – Intelligent distributed surveillance systems*, 152(2) :205–212, 2005.

[MG00] M.A. Mohamed and P. Gader. Generalized hidden Markov models - part i : Theoretical frameworks. *IEEE Trans. on Fuzzy Systems*, 8(1) :67–81, 2000.

[MG01] T.B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Undertanding*, 81 :231–268, 2001.

[MHK06] T.B. Moeslund, A. Hilton, and V. Kruger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104 :90–126, 2006.

[MQD07] D. Mercier, B. Quost, and T. Denoeux. Refined modeling of sensor reliability in the belief function framework using contextual discounting. *To appear in Information Fusion Journal*, 2007.

[MSQ03] S. Mingsheng, S. Shixin, and W. Qingxian. An efficient parallel scheduling algorithm of dependent task graphs. In *Int. Conf. on Parallel and Distributed Computing, Applications and Technologies*, pages 595–598, 2003.

[Mur95] K. Murphy. Passively learning finite automata. Technical report, Santa Fe Institute, available at `http://www.citeseer.ist.psu.edu/murphy95passively.html`, 1995.

[Mur02] K. P. Murphy. *Dynamic Bayesian Networks : Representation, inference and learning*. PhD thesis, UC Berkeley (CSD), 2002.

[MZ03] Y.-F. Ma and H.-J. Zhang. Motion pattern-based video classification retrieval. *EURASIP journal on Applied Signal Processing*, 2 :199–208, 2003.

[NHB04] R. Nevatia, , J. Hobbs, and B. Bolles. An ontology for video event representation. In *IEEE Proc. on Computer Vision and Pattern Recognition*, volume 7, page 119, 2004.

[NLP+02] A. Nefian, L. Liang, X. Pi, L. Xiaoxiang, C. Mao, and K. Murphy. A coupled HMM for audio-visual speech recognition. In *Int. Conf. on Acoustics, Speech and Signal Processing*, 2002.

[NR02] C.W. Ng and S. Ranganath. Real-time gesture recognition system and application. *Image and Vision Computing*, 20 :993–1007, 2002.

[NR03] J.M. Nigro and M. Rombaut. Idres : A rule-based system for driving situation recognition with uncertainty management. *Information Fusion*, 4 :309–317, 2003.

[OB95] J.M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Jour. of Visual Communication and Image Representation*, 6(4) :348–365, 1995.

[OB97] J.M. Odobez and P. Bouthemy. *Video Data Compression for Multimedia Computing*, chapter 8, pages 283–311. Kluwer Academic, 1997.

[OND+05] I. Otsuka, K. Nakane, A. Divakaran, K. Hatanaka, and M. Ogawa. A highlight scene detection and video summarization system using audio feature for a personal video recorder. *IEEE Transactions on Consumer Electronics*, 51(1) :112–116, 2005.

[OSJM07] T. Oliver, B. Schmidt, Y. Jakop, and D.L. Maskell. High speed biological sequence analysis with hidden Markov models on reconfigurable platforms. *IEEE Trans. on Information Theory in Biomedicine*, 2007. To appear (DOI : 10.1109/TITB.2007.904632).

[Pan06] C. Panagiotakis. *Motion Analysis and Modeling for Activity Recognition and 3-D Animation based on Geometrical and Video Processing Algorithms*. PhD thesis, University of Crete, 2006.

[PB98] C. Pinhanez and A. Bobick. Human action detection using pnf propagation of temporal constraints. In *IEEE Proc. on Computer Vision and Pattern Recognition*, June 1998.

[PBPY02] G. Piriou, P. Bouthemy, N. Peyrard, and J.F. Yao. Probabilistic models of image motion for recognition of dynamic content in video. In *Int. Workshop on Computer Vision and Image Analysis*, volume 11, Las Palmas de Gran Canaria, Spain, 2002.

[PBY06] G. Piriou, P. Bouthémy, and J.-F. Yao. Recognition of dynamic video contents with global probabilistic models of visual motion. *IEEE Trans. on Image Processing*, 15(11) :3418–3431, 2006.

[PCD02] K.A. Peker, R. Cabasson, and A. Divakaran. Rapid generation of sports highlights using MPEG-7 motion activity descriptor. *Jour. of Visual Communication and Image Representation*, 15 :265–284, 2002.

[Pie03] W. Pieczynski. Modèles de Markov en traitement d'images. *Traitement du signal*, 20(3) :255–278, 2003.

[Pie07] W. Pieczynski. Multisensor triplet Markov chains and theory of evidence. *Int. Jour. of Approximate Reasoning*, 45 :1–16, 2007.

[Pir05] G. Piriou. *Modélisation statistique du mouvement dans des séquences d'images pour la reconnaissance de contenus dynamiques*. PhD thesis, Université de Rennes 1, Rennes, France, Décembre 2005.

[PR03] M. Pantic and L.J.M. Rothkrantz. Toward an affect-sensitive multimodal human-computer interaction. *Proc. of the IEEE*, 91(9) :1370–1390, 2003.

[PRCM99] V. Pavlovic, J.M. Rehg, T.J. Cham, and K.P. Murphy. A dynamic bayesian network approach to figure tracking using learned dynamic models. In *Int. Conf. on Computer Vision*, pages 94–101, Corfu, Greece, Sept. 1999.

[PRT+06] C. Panagiotakis, E. Ramasso, G. Tziritas, M. Rombaut, and D. Pellerin. Shape-motion based athlete tracking for multilevel action recognition. In F.J. Perales and R.B. Fisher, editors, *Int. Conf. on Articulated Motion and Deformable Objects*, pages 385–394. Springer-Verlag, 2006.

[PRT+07] C. Panagiotakis, E. Ramasso, G. Tziritas, M. Rombaut, and D. Pellerin. Automatic people detection and counting for athletic videos classification. In *IEEE Int. Conf. on Advanced Video and Signal based Surveillance, special session on Vision-based gesture and human action recognition, by I. Patras and E. Hancock*, London, United Kingdom, 2007. Accepted.

[Quo06] Benjamin Quost. *Combinaison de classifieurs binaires dans le cadre de la théorie des fonctions de croyance*. PhD thesis, Université de Technologie de Compiègne, November 2006.

[Rab89] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77 :257–285, 1989.

[RCKL07] M.-C. Roh, B. Christmas, J. Kittler, and S.-W. Lee. Gesture spotting for low-resolution sports video annotation. *Pattern Recognition*, 2007. To appear, doi : 10.1016/j.patcog.2007.07.013.

[RD06] P.F. Riley and J.C. Decker. Analysis architecture of a mobile sports replay system. In *International Conference on Advanced Information Networking and Applications*, 2006.

[RDO06] R. Radhakrishnan, A. Divakaran, and I. Otsuka. Sports program boundary detection. In *ICME*, pages 1621–1624, 2006.

[RFZ07] D. Ramanan, D.A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(1) :65–81, January 2007.

[RJD99] M. Rombaut, I. Jarkass, and T. Denoeux. State recognition in discrete dynamical systems using Petri nets and Evidence theory. In *Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 352–361, 1999.

[RLRNJ00] M. Rombaut, S. Loriette-Rougegrez, J.M. Nigro, and I. Jarkass. Temporal sequence recognition using uncertain sensor data. In *Int. Conf. on Information Fusion*, 2000.

[RPR06] E. Ramasso, D. Pellerin, and M. Rombaut. Belief Scheduling for the recognition of human action sequence. In *Int. Conf. on Information Fusion*, Florence, Italia, 2006.

[RPRP07] E. Ramasso, C. Panagiotakis, M. Rombaut, and D. Pellerin. Human action recognition in videos based on the transferable belief model - application to athletics jumps. *Pattern Analysis and Applications Journal*, 2007. Accepted, doi :10.1007/s10044-007-0073-y.

[RRP07a] E. Ramasso, M. Rombaut, and D. Pellerin. Algorithmes forward, backward et viterbi dans le cadre du modèles des croyances transférables - application à la classification de vidéos. In *21th colloque GRETSI : Traitement du signal et des Images*, Troyes, France, 11-14 Septembre 2007.

[RRP07b] E. Ramasso, M. Rombaut, and D. Pellerin. Forward-backward-viterbi procedures in the transferable belief model for state sequence analysis using belief functions. In *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*,

volume Lecture Notes in Computer Sciences, Hammamet, Tunisia, Oct. 31-Nov. 2 2007. Springer-Verlag. Accepted.

[RRP07c] E. Ramasso, M. Rombaut, and D. Pellerin. State filtering and change detection using TBM conflict - application to human action recognition in athletics videos. *IEEE Trans. on Circuits and Systems for Video Technology*, 17(7) :944–949, 2007. Accepted for publication.

[RS04] B. Ristic and Ph. Smets. Kalman filters for tracking and classification and the Transferable Belief Model. In *Int. Conf. on Information Fusion*, 2004.

[RS05a] B. Ristic and Ph. Smets. Target classification approach based on the belief function theory. *IEEE Trans. Aerospace and Electronic Systems*, 41(2), 2005.

[RS05b] B. Ristic and Ph. Smets. Target identification using belief functions and implication rules. *IEEE Trans. Aerospace and Electronic Systems*, 41(3) :1097–1102, 2005.

[RS06] B. Ristic and Ph. Smets. The TBM global distance measure for the association of uncertain combat ID declarations. *Information Fusion*, 7 :276–284, 2006.

[RZ02] M. Rombaut and Y.M. Zhu. Study of Dempster-Shafer theory for image segmentation applications. *Image and Vision Computing*, 20(1) :15–23, 2002.

[SB06] L. Shen and L. Bai. A review on gabor wavelets for face recognition. *Pattern Analysis and Application Journal*, 9 :273–292, 2006.

[Sch04] J. Schubert. Clustering belief functions based on attracting and conflicting metalevel evidence using Potts spin mean field theory. *Information Fusion*, 5(4) :309–318, 2004.

[Sch06] T.B. Schon. Estimation of non-linear dynamic system – theory and applications. Technical Report ISBN 91-85497-03-7, ISSN 0345-7524, Department of Electrical Engineering, Linköpings universitet, Sweden, 2006.

[Sha48] C.E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27 :379–423,623–656, July-October 1948.

[Sha76] G. Shafer. *A mathematical theory of Evidence*. Princeton University Press, Princeton, NJ, 1976.

[She06] L.M. Sheppard. Visual effects and video analysis lead to olympics victories. *IEEE Computer Graphics and Applications*, 26(2) :6–11, 2006.

[SK94] Ph. Smets and R. Kennes. The Transferable Belief Model. *Artificial Intelligence*, 66(2) :191–234, 1994.

[SKG06] T.B. Schon, R. Karlsson, and F. Gustafsson. The marginalized particle filter in practice. In *IEEE Aerospace Conference*, USA, March 2006.

[SLS03] N. Sebe, M. Lew, and A.W.M Smeulders. Editorial introduction : Video retrieval and summarization. *Computer Vision and Image Understanding*, 92 :141–146, 2003.

[Sme78] Ph. Smets. *Un modèle mathématico-statistique simulant le processus du diagnostic médical*. PhD thesis, Université Libre de Bruxelles, Brussels, Belgium, 1978. (in French).

[Sme92] Ph. Smets. The nature of the unnormalized beliefs encountered in the Transferable Belief Model. In *Proceedings of the 8th Annual Conference on Uncertainty in Artificial Intelligence*, pages 292–297, 1992.

[Sme93] Ph. Smets. Beliefs functions : The Disjunctive Rule of Combination and the Generalized Bayesian Theorem. *Int. Jour. of Approximate Reasoning*, 9 :1–35, 1993.

[Sme94a] Ph. Smets. *Advances in the Dempster-Shafer Theory of Evidence - What is Dempster-Shafer's model ?*, pages 5–34. Wiley, 1994.

[Sme94b] Ph. Smets. Belief induced by the knowledge of some probabilities. In *Uncertainty in artificial intelligence*, pages 523–530. Morgan Kaufmann, 1994.

[Sme95] Ph. Smets. The canonical decomposition of a weighted belief. In *In Int. Joint Conf. on Artificial Intelligence*, pages 1896–1901, San Mateo, Ca, 1995. Morgan Kaufman.

[Sme98] Ph. Smets. The application of the Transferable Belief Model to diagnostic problems. *Int. Jour. of Intelligent Systems*, 13 :127–158, 1998.

[Sme00a] Ph. Smets. Belief functions and the Transferable Belief Model. Available on `www.sipta.org/documentation/belief/belief.ps`, 2000.

[Sme00b] Ph. Smets. Quantified possibility theory seen as an hyper cautious transferable belief model. In Cepadues-Editions, editor, *Rencontres francophones sur les logiques floues et ses applications*, pages 343–353, La Rochelle, France, 2000.

[Sme00c] Ph. Smets. Quantitative epistemic Possibility Theory seen as an hyper-cautious Transferable Belief Model. Technical Report 18, IRIDIA, 2000.

[Sme02a] Ph. Smets. The application of the matrix calculus to belief functions. *Int. Jour. of Approximate Reasoning*, 31(1-2) :1–30, 2002.

[Sme02b] Ph. Smets. Decision making in a context where uncertainty is represented by belief functions. In *Belief Functions in Business Decisions*, pages 17–61. Physica-Verlag, 2002.

[Sme05a] Ph. Smets. Analyzing the combination of conflicting belief functions. *Information Fusion*, 8(4) :387–412, October 2005.

[Sme05b] Ph. Smets. Belief functions on real numbers. *Int. Jour. of Approximate Reasoning*, 40 :181–223, 2005.

[Sme05c] Ph. Smets. Decision making in the TBM : The necessity of the pignistic transformation. *Int. Jour. of Approximate Reasoning*, 38 :133–147, 2005.

[Sme05d] Ph. Smets. Managing deceitful reports with the Transferable Belief Model. *Int. Conf. on Information Fusion*, 2, 2005.

[Smi03] S.W. Smith. Humans in the loop : Human-computer interaction and security. *IEEE Security and Privacy*, pages 75–79, 2003.

[SOMM04] D.A. Sadlier, N. O'Connor, N. Murphy, and S. Marlow. A framework for event detection in field-sports video broadcasts based on SVM generated audio-visual feature model. case-study : Soccer video. In *IWSSIP*, 2004.

[Str04] E. Straszecka. Medical knowledge representation in terms of IF-THEN rules and the Dempster-Shafer theory. In L. Rutkowski et al., editor, *ICAISC*, volume LNAI 3070, pages 1056–1061, 2004.

[SW05] C.G.M. Snoek and M. Worring. Multimedia event-based video indexing using time intervals. *IEEE Trans. on Multimedia*, 7(4), 2005.

[TCP04] D. Tjondronegoro, Y.-P.P. Chen, and B. Pham. Integrating highlights for more complete sports video summarization. *IEEE Multimedia*, 2004.

[TDX+06] W. Tong, L. Duan, C. Xu, Q. Tian, and H. Lu. Local motion analysis and its application in video based swimming style recognition. In *ICPR*, pages 1258–1261, 2006.

[TNB07] Y. Takahashi, N. Nitta, and N. Babaguchi. User and device adaptation for sports video content. In *ICME*, pages 1051–1054, 2007.

[Val01] L. Valet. *Un système flou de fusion coopérative : application au traitement d'images naturelles*. PhD thesis, Université de Savoie, LISTIC, Annecy, December 2001.

[Val03] L. Valet. A fuzzy rule-based interactive fusion system for seismic data analysis. *Information Fusion*, 4(2) :123–133, 2003.

[Van03] P. Vannoorenberghe. Un état de l'art sur les fonctions de croyance appliquées au traitement de l'information. *Revue I3*, 3(1) :9–45, 2003.

[VBT03] V.T Vu, F. Brémond, and M. Thonnat. Automatic video interpretation : a novel algorithm for temporal scenario recognition. In *IJCAI*, Acapulco, Mexico, 2003.

[VD01] P. Vannoorenberghe and T. Denoeux. Diagnostic de la pollution atmosphérique par une approche rdf utilisant les fonctions de croyance. In *Colloque Automatique et Environnement*, Saint-Etienne, Juillet 2001.

[VD02] P. Vannoorenberghe and T. Denoeux. Handling uncertain labels in multiclass problems using belief decision trees. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 1919–1926, 2002.

[VH06] F. Valente and H. Hermansky. Combination fo acoustic classifiers based on Dempster-Shafer theory of evidence. Technical Report 06-61, IDIAP, 2006.

[VHS+02] M. Vihola, M. Harju, P. Salmela, J. Suontausta, and J. Savela. Two dissimilarity measures for HMMs and their application in phoneme model clustering. In *ICASSP*, volume 1, pages 933–936, 2002.

[VMBK01] L. Valet, G. Mauris, P. Bolon, and N. Keskes. Fuzzy seismic facies detection with interactive defuzzificationmethod. In *Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, volume 1, pages 453–458, 2001.

[VS05] P. Vannoorenberghe and Ph. Smets. Partially supervised learning by a Credal EM approach. In *Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, volume 3571, pages 956–967, 2005.

[VSD07] B.L. Van, S. Garcia Salicetti, and B. Dorizzi. On using the viterbi path along with HMM likelihood information for online signature verification. *IEEE Trans. Systems, Man and Cybernetics B*, 37(5) :1237–1247, 2007.

[VT00] R.C. Veltkamp and M. Tanase. Content-based image retrieval systems : A survey. Technical Report UU-CS-2000-34, GIVE-lab, University of Utrecht, NL, 2000.

[VTW06] T. Vercauteren, A.L. Toledo, and X. Wang. Batch and sequential bayesian estimators of the number of active terminals in an ieee 802.11 network. *IEEE Trans. on Signal Processing*, 55(2) :437–450, 2006.

[Vu04] V.T. Vu. *Temporel scenario for automatic video interpretation*. PhD thesis, INRIA, Sophia-Antipolis, 2004.

[VV05] M. Valera and S.A. Velastin. Intelligent distributed surveillance systems : A review. *IEEE Int. Conf. on Vision, Image and Signal Processing*, 152(2) :192–204, 2005.

[WB01] A. Wilson and A. Bobick. Hidden Markov models for modeling and recognizing gesture under variation. *Int. Jour. of Pattern Recognition and Artificial Intelligence*, 15(1) :123–160, 2001.

[WCK06] K.R. Wheeler, M.H. Chang, and K.H. Knuth. Gesture-based control and EMG decomposition. *IEEE Trans. on Systems, man and cybernetics C*, 36(4) :503–514, 2006.

[WCX+06] J. Wang, E. Chng, C. Xu, H. Lu, and X. Tong. Identify sports videos shots with happy or sad emotions. In *ICME*, pages 877–880, 2006.

[WCX+07] J. Wang, E. Chng, C. Xu, H. Lu, and Q. Tian. Generation of personalized music sports video using multimodal cues. *IEEE Trans. on Multimedia*, 9(3) :576–588, 2007.

[WF05] I.H. Witten and E. Frank. *Data-mining : practical machine learning tools and techniques*. 2nd edition. Morgan Kaufman, 2005.

[WHT03] L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3) :585–601, 2003.

[WS03] J. Wang and S. Singh. Video analysis of human dynamics-a survey. *Real-Time Imaging*, 9(5) :321–346, 2003.

[WXC06] J. Wang, C. Xu, and E. Chng. Automatic sports video genre classification using pseudo-2d-HMM. In *ICPR*, 2006.

[WYX05] K. Wan, X. Yan, and C. Xu. Automatic mobile sports highlights. In *ICME*, July 2005.

[XG03] T. Xiang and S. Gong. Discovering Bayesian causality among visual events in a complex outdoor scene. In *IEEE on Advanced Video and Signal based Surveillance*, pages 177–182, 2003.

[XHS96] H. Xu, Y-T. Hsia, and Ph. Smets. Transferable Belief Model for decision making in the Valuation-Based Systems. *IEEE Trans. on Systems, Man and Cybernetics*, 26, 1996.

[XMZY05] G. Xu, Y.-F. Ma, H.-J. Zhang, and S.-Q. Yang. An HMM-based framework for video semantic analysis. *IEEE Trans. on Circuits and Systems for Video Technology*, 15(11) :1422–1433, 2005.

[XRDH05] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T.S. Huang. Highlights extraction from sports video based on an audio-visual marker detection framework. In *ICME*, July 2005.

[XS94] H. Xu and Ph. Smets. Evidential reasoning with conditional belief functions. *Uncertainty in Artificial Intelligence*, 94 :598–606, 1994.

[XZT+06] Z. Xiong, X.S. Zhou, Q. Tian, Y. Rui, and T.S. Huang. Semantic retrieval of video. *IEEE Signal Processing Magazine*, pages 18–27, 2006.

[Yag87] R.R. Yager. On the Dempster-Shafer framework and new combination rules. *Information Sciences*, 41 :93–137, 1987.

[YHYC05] X. Yu, T.S. Hay, X. Yan, and E. Chng. A player-possession acquisition system for broadcast soccer video. In *ICME*, July 2005.

[YLL+03] X. Yu, H.W. Leonga, J.H. Lim, Q. Tian, and Z. Jiang. Team possession analysis for broadcast soccer video based on ball trajectory. In *PCM*, pages 1811–1815, 2003.

[YLSL07] J. You, G. Liu, Li S., and H. Li. A multiple visual models based perceptive analysis framework for multilevel video summarization. *IEEE Trans. on Circuits and Systems for Video Technology*, 17(3) :273–285, 2007.

[YLXT06] X. Yu, H.W. Leong, C. Xu, and Q. Tian. Trajectory-based ball detection and tracking in broadcast soccer video. *IEEE Trans. on Multimedia*, 8(6) :1164–1178, 2006.

[YPL07] H.-D. Yang, A-Y. Park, and S.-W. Lee. Gesture spotting and recognition for human-robot interaction. *IEEE Trans. on Robotics*, 23(2) :256–270, 2007.

[YS05] A. Yilmaz and M. Shah. Recognizing human actions in videos acquired by uncalibrated moving cameras. In *ICCV*, Beijing, China, October 2005.

[YSM01] B. Yaghlane, Ph. Smets, and K. Mellouli. Belief function independence : I. The marginal case. *Int. Jour. of Approximate Reasoning*, 29 :47–70, 2001.

[YSM02a] B. Yaghlane, Ph. Smets, and K. Mellouli. Belief function independence : II. The conditionnal case. *Int. Jour. of Approximate Reasoning*, 31(1) :31–75, October 2002.

[YSM02b] B. Yaghlane, Ph. Smets, and K. Mellouli. Independence concept for belief functions. In *Technologies for constructing intelligent systems : Tools*. Physica-Verlag, 2002.

[YSM03] B. Ben Yaghlane, Ph. Smets, and K. Mellouli. Directed evidential networks with conditionnal belief functions. In *Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 291–305, 2003.

[YYHL04] X. Yu, X. Yan, T.S. Hay, and H. W. Leong. 3D reconstruction and enrichment of broadcast soccer video. In *ACM MM*, pages 260–263, 2004.

[ZBTV07] N. Zouba, F. Bremond, M. Thonnat, and V.T. Vu. Multi-sensors analysis for everyday activity monitoring. In *Int. Conf. on Sciences of Electronic, Technologies of Information and Telecommunications*, pages 1–9, Tunisia, 2007.

[ZCDZ07] B. Zhang, W. Chen, W. Dou, and Y.-J Zhang. Content-based table tennis games highlight detection utilizing audiovisual clues. In *Int. Conf. on Image and Graphics*, pages 833–838, 2007.

[ZD98a] L.M. Zouhal and T. Denoeux. An adaptive K-NN rule based on Dempster-Shafer theory. In *Proc. of the 6th Int. Conf. on Computer Analysis of Images and Patterns*, pages 310–317, Prague, September 1998. Springer-Verlag.

[ZD98b] L.M. Zouhal and T. Denoeux. An evidence-theoretic K-NN rule with parameter optimization. *IEEE Trans. on Systems, Man and Cybernetics*, 28 :262–271, 1998.

[ZGPB05] D. Zhang, D. Gatica-Perez, and S. Bengio. Semi-supervised meeting event recognition with adapted HMMs. In *Proc. IEEE ICME*, 2005.

[ZHX+07] G. Zhu, Q. Huang, C. Xu, L. Xing andW. Gao, and H. Yao. Human behavior analysis for highlight ranking in broadcast racket sports video. *IEEE Trans. on Multimedia*, 9(6) :1167–1182, 2007.

[ZXR+07] Y. Zhang, C. Xu, Y. Rui, J. Wang, and H. Lu. Semantic event extraction from basketball games using multimodal analysis. In *ICME*, July 2007.

# Liste des publications

## Journaux internationaux

E. Ramasso and C. Panagiotakis and M. Rombaut and D. Pellerin, Human action recognition in videos based on the Transferable Belief Model - Application to athletics jumps, *Pattern Analysis and Applications Journal*, Accepted, doi :10.1007/s10044-007-0073-y, 2007.

E. Ramasso and M. Rombaut and D. Pellerin , State filtering and change detection using TBM conflict - Application to human action recognition in athletics videos, *IEEE Trans. on Circuits and Systems for Video Technology*, 17(7) :944-949, 2007.

L. Valet and E. Ramasso and S. Teyssier, Quality evaluation of insulating parts by fusion of classifiers issued from tomographic images, *Information Fusion, Special issue on Image Fusion in Industrial Applications*, Accepted, doi :10.1016/j.inffus.2006.05.002, 2006.

## Conférences internationales avec comité de lecture

E. Ramasso and M. Rombaut and D. Pellerin, Forward-Backward-Viterbi procedures in the Transferable Belief Model for state sequence analysis using belief functions, *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, Lecture Notes in Computer Sciences, Hammamet, Tunisia, Oct. 31-Nov. 2, 2007, Springer-Verlag.

C. Panagiotakis and E. Ramasso and G. Tziritas and M. Rombaut and D. Pellerin, Automatic People Detection and Counting for Athletic Videos Classification, *IEEE Int. Conf. on Advanced Video and Signal based Surveillance, special session on Vision-based gesture and human action recognition*, London, United Kingdom, 2007.

E. Ramasso and D. Pellerin and M. Rombaut, Belief Scheduling for the recognition of human action sequence, *Int. Conf. on Information Fusion*, Florence, Italia, 2006.

C. Panagiotakis and E. Ramasso and G. Tziritas and M. Rombaut and D. Pellerin, Shape-motion based athlete tracking for multilevel action recognition, *Int. Conf. on Articulated Motion and Deformable Objects*, pages 385-394, 2006, Springer-Verlag. **Award for the most suitable work for commercialization**.

E. Ramasso and M. Rombaut and D. Pellerin, A Temporal Belief Filter improving human action recognition in videos, *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, volume 2, pages 141-144, May 14-19, Toulouse, France, 2006.

E. Ramasso and D. Pellerin and M. Rombaut, Reconnaissance de sauts d'athlètes dans les vidéos : fusion du mouvement de caméra et de la trajectoire de points caractéristiques, *20th colloque GRETSI : Traitement du signal et des Images*, volume 2, pages 1165-1168, Louvain-la-Neuve, Belgique, 6-9 Septembre, 2005.

E. Ramasso and D. Pellerin and C. Panagiotakis and M. Rombaut and G. Tziritas and W. Lim, Spatio-temporal information fusion for human action recognition in videos, *13th European Signal Processing Conference*, Antalya, Turkey, September 4-8, 2005.

E. Ramasso and L. Valet and S. Teyssier, Classification fusion by decision templates for insulating part control, *IEEE/ISIF 8th Int. Conf. on Information Fusion*, pages 423-428, volume 1, Philadelphia, USA, July 25-29, 2005.

E. Ramasso and L. Valet and S. Teyssier, Partial decision aggregation for electrotechnical parts quality control, *IEEE Instrumentation and Measurement Technology Conference*, volume 3, pages 1628-1632, Ottawa, Canada, May 17-19, 2005.

## Conférences francophones avec comité de lecture

E. Ramasso and M. Rombaut and D. Pellerin, L'ordonnanceur crédibiliste pour la reconnaissance de séquences d'états dans le cadre du Modèle des Croyances Transférables, *Rencontres francophones sur la Logique Floue et ses Applications*, Nimes, France, 22-23 Novembre, 2007.

E. Ramasso and M. Rombaut and D. Pellerin, Algorithmes Forward, Backward et Viterbi dans le cadre du Modèles des Croyances Transférables - Application à la classification de vidéos, *21th colloque GRETSI : Traitement du signal et des Images*, Troyes, France, 11-14 Septembre, 2007.

E. Ramasso and M. Rombaut and D. Pellerin, Un filtre temporel crédibiliste pour la reconnaissance d'actions humaines dans les vidéos, *Rencontres francophones sur la Logique Floue et ses Applications*, Toulouse, France, 19-20 Octobre, 2006.

M. Rombaut and D. Pellerin and E. Ramasso, Vers la reconnaissance de motifs temporels, *5èmes journées sur l'Extraction et la Gestion des Connaissances, Atelier 7 : Extraction de motifs temporels pour la détection en ligne de situations critiques*, 18 janvier, Paris, France, 2005.

## Autres communications

M. Rombaut, E. Ramasso and F. Maussang, La théorie des fonctions de croyance : une alternative aux probabilités pour la perception et la fusion de données, *Revue de l'Electricité et de l'Electronique*,

volume 8, p. 51-60, Septembre 2006, ISSN : 1265-6534.

D. Anache-Ménier, G. Jourdan, E. Ramasso, E. Rosembum, A. Serna, PJ Zermatten, Sculpture : volumes et matériaux, *Article CIES*, Exposition « art et science » au musée de Grenoble, 14-27 Mars, 2007.

## Articles soumis

E. Ramasso, C. Panagiotakis, M. Rombaut, D. Pellerin, Belief Scheduler for sequential data analysis based on TBM – Application to activity recognition in athletics videos. Soumis à *IEEE Trans. on System, Man and Cybernetics*, Avril 2007.

E. Ramasso, C. Panagiotakis, M. Rombaut, D. Pellerin and G. Tziritas, Human Shape-Motion Analysis In Athletics Videos for Coarse To Fine Action/Activity Recognition Using Transferable Belief Model. Soumis à *Electronic Letters on Computer Vision and Image Analysis*, Juin 2007.

C. Panagiotakis, E. Ramasso, G. Tziritas, M. Rombaut and D. Pellerin, Shape-based Individual/Group Detection for Sport Videos Categorization. Soumis à *Int. Journal on Pattern Recognition and Artificial Intelligence*, Juillet 2007, accepté après révisions mineures.

# Liste des tableaux


2.1 Exemple de BBA $m^{\Omega}$ exprimée sur un cadre de discernement $\Omega = \{\omega_1, \omega_2, \omega_3\}$. Le cas *crédibiliste* (fonction de masses de croyance quelconque) permet de couvrir tous les cas ($6^e$ colonne), i.e. de la connaissance précise et certaine (colonne 2) à l'ignorance totale (colonne 7). Inspiré de [VS05]. . . . . 22
2.2 Utilisation des fonctions de croyance dans le cadre de nos travaux. . . . . 23
2.3 Résultat de la transformation de la BBA donnée dans la $6^e$ colonne du tableau 2.1 en plausibilité ($pl$), crédibilité ($bel$), implicabilité ($b$) et communalité ($q$). . . . . 24
2.4 Base de connaissance liant l'état du temps au jour $j$ connaissant l'état du temps au jour $j-1$. . . . . 30
2.5 Base de connaissance probabiliste $P^{\Omega_j}(.|\omega_{j-1})$ issue de la transformation pignistique des distributions du tableau 2.4. . . . . 31
2.6 Les plausibilités a posteriori $pl^{\Omega_{j-1}}[\omega_j]$ définies sur $\Omega_{j-1}$ conditionnellement aux hypothèses et sous-ensembles d'hypothèses de $\Omega_j$ obtenues par le GBT. . . . . 33
2.7 Distributions de probabilités $P^{\Omega_{j-1}}(.|\omega_j)$ au jour $j-1$ conditionnellement aux éléments de $\Omega_j$. . . . . 33
2.8 Les masses définies sur $\Omega_j$ conditionnellement aux hypothèses et sous-ensembles d'hypothèses de $\Omega_{j-1}$ obtenues par la DRC. . . . . 35
2.9 Les règles de combinaison et leur contexte d'utilisation [Den07]. . . . . 45

4.1 Indices de performance : rappel ($R$), précision ($P$) et mesure $F_1$ en $\%$ sur la détection des actions fournies par les sources (ligne 1), après application du filtre TCF (ligne 2) et après application de l'ordonnanceur CSS (ligne 3). . . . . 82

5.1 Matrice de transition probabiliste $P^{\Omega_t}(.|\omega_{t-1})$ obtenue par estimation [Rab89]. . . . 99
5.2 Résumé des principaux mécanismes des HMM. . . . . 106
5.3 Notations probabilistes (colonne « Proba. ») et crédibilistes (colonne « TBM ») des différentes variables et algorithmes des HMM avec les équations correspondantes. . 108
5.4 Matrice de transition du CrHMM. . . . . 112
5.5 Indices de performance : rappel ($R$), précision ($P$) et mesure $F_1$ en $\%$ sur la détection des actions par le décodage de Viterbi probabiliste (ligne 1), par maximisation de la probabilité $\gamma_t$ a posteriori (ligne 2), par le décodage de Viterbi crédibiliste (ligne 3) et par maximisation de la probabilité pignistique de $q_{\gamma}^{\Omega_t}$ a posteriori. . . . . 119

6.1 Description de la base en terme de nombre d'images, où $N_v$ correspond au nombre de vidéos. . . . . 136

6.2 Conditions expérimentales : association des méthodes de modélisation et des algorithmes de reconnaissance. . . . . 155
6.3 Reconnaissance des actions dans les *sauts en hauteur* par le modèle CSS+EX (gain apporté par le CSS pour chaque action par rapport aux observations EX : $+14\%$, $+5\%$, $+21\%$ et $+32\%$). . . . . 158
6.4 Reconnaissance des actions dans les *sauts à la perche* par le modèle CSS+EX (gain apporté par le CSS pour chaque action par rapport aux observations EX : $+23\%$, $+12\%$, $-4\%$ et $+20\%$). . . . . 158
6.5 Reconnaissance des actions dans les *sauts en hauteur* avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS. 158
6.6 Reconnaissance des actions dans les *sauts en hauteur* avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités. . . . . 158
6.7 Reconnaissance des actions dans les *sauts en longueur* avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS. 159
6.8 Reconnaissance des actions dans les *sauts en longueur* avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités. . . . . 159
6.9 Reconnaissance des actions dans les *sauts à la perche* avec les mélanges gaussiens (GMM, première série de tableaux) et le modèle de distance (EDC, seconde série de tableaux) en entrée. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par les deux types de modèle. Troisième colonne : application du modèle de Markov caché probabiliste (HMM). . . . . 159
6.10 Reconnaissance des actions dans les *sauts à la perche* avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités. . . . . 160

6.11 Reconnaissance des actions dans les *triple sauts* avec les mélanges gaussiens (GMM) en entrée. Première colonne : résultat de la détection avec la modélisation GMM sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM). Troisième colonne : application du modèle de Markov caché probabiliste (HMM). Quatrième colonne : application de l'ordonnanceur CSS. . . . . 160
6.12 Reconnaissance des actions dans les *triples sauts* avec le modèle de distance (EDC) en entrée. Première colonne : résultat de la détection avec la modélisation EDC sans séquencement. Deuxième colonne : application du modèle de Markov caché crédibiliste (CrHMM) sur les croyances fournies par le modèle EDC. Troisième colonne : application du modèle de Markov caché probabiliste (HMM) après transformation des croyances fournies par le modèle EDC en probabilités. . . . . 160
6.13 Classification par HMM probabiliste (gauche, taux de bonne classification de $60\%$), par HMM crédibiliste (centre, taux de bonne classification de $60\%$) et par l'ordonnanceur CSS (droite, taux de bonne classification de $68\%$) lorsque les entrées sont fournies par des mélanges gaussiens (GMM). . . . . 168
6.14 Classification par CSS+EX. . . . . 168

# Table des figures


2.1 Représentation abstraite des mécanismes dans le Modèle des Croyances Transférables. . . . . 17

2.2 Fusion d'information : des sources de croyance émettent une opinion pondérée sur l'état du système observé. Les différentes sources sont combinées afin de synthétiser la compréhension du système et améliorer la décision sur l'état du système après combinaison. . . . . 18

2.3 Modèle graphique représentant le lien de causalité entre deux variables à valeurs discrètes dans $\Theta$ et $\Omega$. Le lien de causalité est traduit par un conditionnement $(m^{\Omega}[\theta], \forall \theta \subseteq \Theta)$. . . . . 36

2.4 Raffinement de cadre de discernement : $\Theta$ est un raffinement de $\Omega$. . . . . 37

3.1 Un résultat de filtrage de fonctions de croyance temporelles bruitées par le TCF proposé. 55

3.2 Principe du filtre temporel crédibiliste TCF. . . . . 56

3.3 Un résultat de filtrage. . . . . 61

3.4 Evolution de la qualité pour le filtrage de la figure 3.3. . . . . 64

3.5 Evolution de la mesure $F_1$ en fonction des couples $(\mathbb{T}_s, \lambda)$ pour la détection des actions *courir*, *sauter* et *chuter* dans un saut en hauteur (mesure moyennée sur $10$ sauts en hauteur). Deux ensembles de résultats sont présentés : détection des actions par la fusion de BBA issues d'une modélisation heuristique basée sur les sous-ensembles flous (haut) et issues de mélanges Gaussiens après apprentissage intensif (bas). Image tirée de [RRP07c]. . . . . 66

3.6 Estimation du seuil d'arrêt : a) données d'apprentissage avec la vérité terrain indiquée par un trait noir, b) le filtrage avec un seuil d'arrêt infini, c) évolution de la somme cumulée du conflit lorsque le seuil d'arrêt est infini et d) le filtrage avec un seuil d'arrêt fixé à 2.1 obtenu comme indiqué à l'exemple 19. . . . . 68

3.7 Un jeu de données d'apprentissage : action « course » dans un saut en hauteur. La vérité terrain est donnée par le trait noir sur l'axe temporel. . . . . 69

3.8 Application du TCF pour le filtrage de fonctions de croyance non binaires. . . . . 71

3.9 Illustrations du grossissement pour le filtrage de fonctions de croyance définies sur des espaces non binaires. . . . . 71

4.1 BPN : évolution de la masse et problème des fausses alarmes. . . . . 77

4.2 Une séquence $S_1$ composée de $4$ états. . . . . 78

4.3 Représentation d'une séquence de 4 états entre deux instants successifs. Le filtre TCF est utilisé pour assurer une cohérence temporelle sur chaque état (sens horizontal) et la machine à états intègre des informations sur la causalité entre les états (sens vertical). 78

4.4 L'activité *saut en hauteur* décrite comme une séquence de $4$ actions. . . . . 79

4.5 Croyances sur les actions avant filtrage, fournies par les sources. Courbe bleue : action vraie, courbe rouge : action fausse, courbe verte : action vraie ou fausse. . . . . 80

4.6 Croyances sur les actions après filtrage. . . . . 81

4.7 Croyances sur les actions après filtrage et prise en compte des liens de causalité par l'ordonnanceur. . . . . 82

4.8 Détection des états dans le cas idéal et dans le cas réel. . . . . 83

4.9 Détection des états dans le cas idéal avant et après traitement par l'ordonnanceur. . . 83

4.10 Préemption : l'état $E_{k+1}$ prend naturellement le modèle « $\mathcal{T}$ : l'état est vrai » tandis que l'état $E_k$ est contraint à utiliser le modèle « $\mathcal{F}$ : l'état est faux ». . . . . 84

4.11 Forçage : l'état $E_k$ utilise naturellement le modèle « $\mathcal{F}$ : l'état est faux » tandis que l'état $E_{k+1}$ est contraint à utiliser le modèle « $\mathcal{T}$ : l'état est vrai ». . . . . 85

4.12 Détection des fausses alarmes : le critère de qualité locale $LQ_{2,1}(T^f)$ représente la qualité de la proposition « *état* $E_2$ *est vrai* » conditionnellement au modèle vrai dans la séquence $S_1$. . . . . 86

4.13 Fausse alarme : une préemption suivie d'un forçage indique une fausse alarme potentielle. . . . . 86

4.14 Inférence réalisée en ligne par agrégation des qualités locales. . . . . 87

4.15 Evolution du critère d'inférence au cours de la reconnaissance de la séquence d'états. 88

4.16 Le saut à la perche de la vidéo 8 présente un angle de vue qui rend difficile la reconnaissance du saut. . . . . 89

4.17 Le saut à la perche de la vidéo 24 présente un angle de vue qui rend difficile la reconnaissance du saut. . . . . 89

4.18 Valeur des critères de reconnaissance de séquences $GQ$ pour $26$ vidéos de *saut à la perche* en utilisant les $4$ ordonnanceurs CSS. . . . . 90

4.19 Valeur des log-vraisemblances en sortie des $4$ HMM (un pour chaque saut) pour la reconnaissance de séquences de $26$ *sauts à la perche*. . . . . 91

4.20 Evolution du critère $GQ$ dans l'ordonnanceur pour la reconnaissance en ligne d'un saut à la perche (vidéo numéro 2 de la figure 4.19). . . . . 91

5.1 Un HMM composé de $K$ états à deux instants consécutifs et des observations *numériques* continues ($5$ sources d'information) évoluant au cours du temps qu'il s'agit de lier aux états. Les unités de ces observations n'ont pas d'importance ici. . . . . 94

5.2 Modèle graphique représentant les propagations *forward* et *backward* entre deux instants et où $\Omega_t$ est l'ensemble des états. . . . . 96

5.3 Propagation des distributions, figure inspirée de [Rab89, MG00]. Les traits continus représentent la propagation *forward*, et les traits pointillés la propagation *backward*. Les expressions des variables du même nom sont rappelées respectivement au-dessus et au-dessous de la figure. . . . . 98

5.4 Résultats du filtrage des (a) vraisemblances (normalisées) sur les états obtenues à l'aide des observations, (b) par la propagation *forward* probabiliste et (c) par la propagation *backward* probabiliste. La figure (d) donne le résultat de la combinaison conjonctive des deux propagations à chaque instant (propagation dans les deux directions). . . . . 100

5.5 Algorithme de Viterbi, inspiré de [Rab89, MG00]. . . . . 101

5.6 Résultats de l'algorithme de Viterbi (a) pour la reconnaissance de la séquence d'états la plus probable à partir des observations de la figure 5.4a. Comparaison avec la détermination du meilleur état (b) en utilisant la maximisation a posteriori (Eq. 5.12) sur les probabilités $\gamma_t$ données à la figure 5.4d. . . . . 102

5.7 Modèle graphique représentant les propagations *forward* et *backward* crédibilistes entre deux instants où $\Omega_t$ est l'ensemble des états. . . . . 109

5.8 Résultats du filtrage des croyances représentées en (a) : par les propagations *forward* (b) et *backward* (c) crédibilistes. La figure (d) donne le résultat de la combinaison conjonctive des deux propagations à chaque instant (propagation dans les deux directions). Un trait gras représente le cas crédibiliste et un trait fin représente le cas probabiliste (pour comparaison). . . . . 113

5.9 Les informations disponibles à $t$ : la distribution $f_\alpha^{\Omega_{t-1}}$ sur les états à $t-1$, les transitions $f_a^{\Omega_t}[s_i^{t-1}]$ et les croyances $f_b^{\Omega_t}$ relatives aux observations. . . . . 114

5.10 Résultats de l'algorithme de Viterbi crédibiliste (a) pour la reconnaissance de la séquence d'états la plus probable à partir des observations de la figure 5.8a. Comparaison avec la détermination du meilleur état (b) en utilisant la maximisation a posteriori (Eq. 5.12) sur les probabilités pignistiques de $q_\gamma^{\Omega_t}$ données à la figure 5.8d. . . . . 119

5.11 Superposition des décodages de Viterbi crédibiliste et probabiliste pour la reconnaissance du saut en hauteur (Figs. 5.8-5.10) sur une vidéo composée de $154$ images. Le saut en hauteur est décrit par une séquence de quatre actions à savoir *courir* (valeur $1$ sur l'axe des ordonnées de la figure), *sauter* (valeur $2$), *chuter* (valeur $3$) et *se relever* (valeur $4$). . . . . 120

5.12 Evolution des critères d'inférence pour le cas crédibiliste (a) et probabiliste (b) pour la reconnaissance du saut en hauteur (Figs. 5.8-5.10). . . . . 120

5.13 Evolution des probabilités pignistiques $\mathbf{BetP}^{\Omega_t}\{m_\delta^{\Omega_t}\}$ calculées à partir des BBA $m_\delta^{\Omega_t}$ propagées dans l'algorithme de Viterbi crédibiliste (courbe de couleur rouge et en gras). Les probabilités $\delta$ dans le cas probabiliste sont aussi illustrées (courbe de couleur bleue en trait fin). . . . . 121

6.1 Contexte de l'application. . . . . 129
6.2 Architecture du système. . . . . 135
6.3 Extraction d'informations sur les actions. . . . . 136
6.4 Quelques images montrant l'hétérogénité de la base. . . . . 137
6.5 Evolution des paramètres de translations horizontale et verticale (en pixels par image) et du zoom de la caméra estimés par *motion2D* pour un *saut en hauteur* avec 4 actions : *courir* sur $[1, 58]$, *sauter* sur $[59, 72]$, *chuter* sur $[73, 90]$ et *se relever* sur $[91, 106]$. 138
6.6 Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en *extérieur* et obtenue à partir d'un site internet (moyenne qualité). . 139
6.7 Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en *intérieur* et obtenue à partir d'un enregistrement analogique (basse qualité). . . . . 140
6.8 Effet de la variation d'illumination sur l'estimation du mouvement dominant dans une vidéo acquise en *extérieur* et obtenue à partir d'une DVD (haute qualité). . . . . 141
6.9 Evolution en pourcent de la variation relative d'intensité lumineuse moyenne entre deux images successives dans (*haut–gauche*) un *saut en hauteur*, (*haut–droite*) un *saut à la perche*, (*bas–gauche*) un *triple saut* et (*bas–droite*) *un saut en longueur*. . . 142
6.10 Discrimination entre des vidéos de sports d'équipe et individuel. . . . . 142
6.11 Résultats de l'algorithme de catégorisation pour un *triple saut*, tiré de [PRT+07]. La première rangée d'images correspond à une partie des images originales, la seconde rangée représente le résultat de l'algorithme. La silhouette apparaît en blanc et son centre de gravité avec un petit carré. Les autres blocs de pixels (en gris) correspondent à du bruit (éliminé automatiquement par l'algorithme). La figure en bas à droite représente la masse de croyance sur le fait que la vidéo correspond à un sport collectif (T, en rouge), un sport individuel (I, en bleu) et le doute entre ces deux hypothèses (en vert). Ici la vidéo est un *triple saut* correctement classé en tant que sport individuel. Les figures en bas à gauche correspondent au nombre d'objets (valeur approximative réelle dont seule la partie entière a un sens) dans la vidéo (première courbe) et à un indice de qualité concernant l'estimation (deuxième courbe). . . . . 144
6.12 Résultats de l'algorithme de catégorisation pour un sprint (100m), tiré de [PRT+07]. Voir la légende de la figure 6.11. Ici la différence vient du fait que la vidéo correspond à un sport collectif. Au départ de la course, les athlètes sont en ligne et la caméra de face donc les $8$ sprinters sont bien détectés. Puis la caméra bouge et son axe optique devient perpendiculaire au mouvement des athlètes. A cause des occultations, seulement deux ou trois athlètes sont détectés. . . . . 145
6.13 Evolution de la masse de croyance sur le fait que la vidéo est un sport individuel fournie par l'algorithme de catégorisation pour une vidéo composée de plusieurs sports collectifs et d'un sport individuel (*saut en hauteur*). L'algorithme fonctionne en ligne et il est capable d'estimer le nombre d'objets mobiles (ici des athlètes) dans la vidéo au fil de l'eau. La zone détectée (de l'image $40$ à l'image $100$ environ) correspond au *saut en hauteur*, la courbe verte représente les transitions entre les sports. Les images originales et les images de la détection sont représentées sur la figure 6.14. . . . . 145

6.14 Images originales et images de la détection de l'algorithme de catégorisation pour une vidéo composée de plusieurs sports collectifs et d'un sport individuel (*saut en hauteur*). Cette figure est associée à la courbe de la figure 6.13. . . . . 146
6.15 Résultat du suivi de points pour un *saut en hauteur*, un *saut en longueur*, un *saut à la perche* et un *triple saut*. . . . . 147
6.16 Illustration de la variation verticale du centre de gravité. En haut est représentée l'évolution au cours du temps de la position (en pixels) verticale du centre de gravité. Les paliers sur la courbe du bas correspondent aux tendances de variation estimés par des modèles linéaires (valeur des coefficients directeurs). Exemple : le centre de gravité a une variation verticale positive importante sur l'intervalle $[60, 100]$. . . . . 148
6.17 Principe du calcul de l'angle et un exemple d'évolution pour un *saut en longueur*, tiré de [RPRP07]. . . . . 149
6.18 Deux modélisations possibles : soit un classifieur par action ou un classifieur pour les quatre actions. Le premier cas est utilisé pour les tests sur l'ordonnanceur CSS et le deuxième cas pour le CrHMM. . . . . 151
6.19 Conversion numérique-symbolique inspirée des sous-ensembles flous : la caractéristique de translation verticale de la caméra ($F_{cvt}$) est transformée en BBA sur l'action *courir*, $m_{cvt}^{\Omega_t^1}$, définie sur $\Omega_t^1 = \{T_t^1, F_t^1\}$. . . . . 153
6.20 Evolution théorique de l'angle pour un *saut en hauteur*, un *saut à la perche* et un *saut en longueur* pour un mouvement de caméra de gauche à droite. . . . . 154
6.21 Evolution de la mesure $F_1$ en fonction du seuil de stop et de la pondération de la somme cumulée du conflit utilisés dans l'ordonnanceur CSS pour des *sauts en hauteur* avec les mélanges gaussiens (GMM) en entrée du système. Ligne 1 : action *courir* et *sauter*, ligne 2 : actions *chuter* et *se relever*. Les points colorés indiquent un même réglage pour les quatre actions dans les *sauts en hauteur* assurant une performance satisfaisante (resp. $70\%$, $80\%$, $80\%$ et $40\%$ pour les actions *courir*, *sauter*, *chuter* et *se relever*). . . . . 157
6.22 Evolution de la mesure $F_1$ en fonction du seuil de stop et de la pondération de la somme cumulée du conflit utilisés dans l'ordonnanceur CSS pour des *sauts à la perche* avec les mélanges gaussiens (GMM) en entrée du système. Ligne 1 : action *courir* et *sauter*, ligne 2 : actions *chuter* et *se relever*. Les points colorés indiquent un même réglage pour les quatre actions dans les *sauts à la perche* assurant une performance satisfaisante (resp. $70\%$, $70\%$, $60\%$ et $25\%$ pour les actions *courir*, *sauter*, *chuter* et *se relever*). . . . . 161
6.23 Erreur de l'ordonnanceur CSS. Une chute est détectée en $[55, 70]$ perturbant ainsi le déroulement de la séquence. . . . . 162
6.24 Les vraisemblances sur les états pour un *saut en hauteur* obtenues à partir des observations. . . . . 164
6.25 Décision sur l'état après application de l'algorithme de Viterbi probabiliste. . . . . 165
6.26 Décision sur l'état après application de l'algorithme de Viterbi crédibiliste. . . . . 165
6.27 Ensemble utilisé pour le conditionnement à chaque instant dans l'équation 5.57 où $Co$, $S$, $Ch$ et $R$ signifient respectivement *courir*, *sauter*, *chuter* et *se relever*. Les symboles $\cup$ représente une union d'actions. . . . . 166

1 Cadres de discernement formant une hiérarchie. Exemple de la catégorisation d'instruments de musique. . . . . 176
2 Les transitions utilisées dans le réseau de Pétri crédibiliste. . . . . 178
3 Probabilités pignistiques sur les places du réseau de Pétri crédibiliste. . . . . 179
4 Interface graphique pour l'estimation du mouvement de caméra. . . . . 180
5 Interface graphique pour la reconnaissance des séquences d'états. . . . . 181

# Résumé

Cette thèse porte sur la problématique de reconnaissance automatique de systèmes dynamiques. Une méthodologie basée sur des modèles de séquences d'états est employée : les états permettent de décrire le système à un instant particulier tandis que des transitions permettent au système d'évoluer au cours du temps. Dans le cadre de la thèse, deux nouvelles méthodes de représentation et de reconnaissance de séquences d'états basées sur le Modèle des Croyances Transférables, modèle non probabiliste de raisonnement incertain basé sur les fonctions de croyance, sont proposées. La première méthode est déterministe et inspirée des travaux en Intelligence Artificielle, la seconde est stochastique et basée sur une généralisation aux fonctions de croyance des modèles de Markov cachés initialement développés dans la théorie des probabilités. Ces algorithmes, dont le cadre formel est générique, ont été intégrés dans un système de reconnaissance de mouvements humains dans les vidéos d'athlétisme que nous avons mis en place en collaboration avec l'Université de Crète dans le cadre du Réseau d'Excellence Européen SIMILAR. Les méthodes de reconnaissance de séquences ont été évaluées sur une base de 74 vidéos et comparées aux modèles de Markov cachés probabilistes.



**Mots clefs :** Séquences d'états, Modèle des Croyances Transférables, Fonction de croyance, Modèle de Markov caché, Fusion d'information, Analyse de mouvements humains, Indexation de vidéos.



# Abstract

This thesis focuses on automatic recognition of dynamical systems. A methodology based on state sequence models has been chosen : states describe the system at particular instants while transitions allow it to evolve along time. In this thesis, two new methods based on Transferable Belief Model, a non-probabilistic model for uncertain reasoning based on belief functions, are proposed : the first one is deterministic and inspired from works in Artificial Intelligence and the second one is stochastic generalizing Hidden Markov Models, initially developed in probability theory, to belief functions. These algorithms, which are generic, have been integrated in a system aiming at recognizing human motions in athletics videos. The system has been set in collaboration with University of Crete within SIMILAR European Network of Excellence. The state sequence recognition method proposed in this thesis have been assessed on a database made of 74 videos and compared to probabilistic Hidden Markov Models.



**Keywords :** State sequence, Transferable Belief Model, Belief function, Hidden Markov model, Information fusion, Human motion analysis, Video indexing.