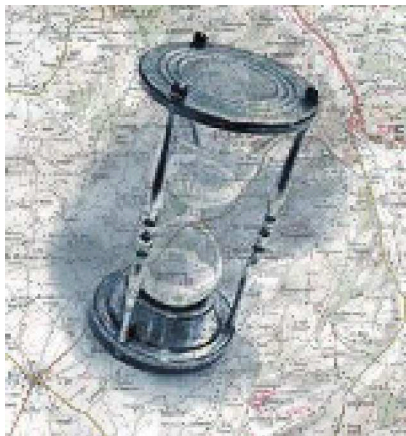


RFIA / Caen, du 19 au 22 janvier 2010

CINQUIÈME ATELIER :

Représentation et raisonnement sur le temps et l'espace (RTE 2010)



Maroua Bouzid
Florence Le Ber
Gérard Ligozat
Odile Papini

Cinquième Atelier "Représentation et Raisonement sur le temps et l'espace"

Caen, 19 janvier 2009

Avant propos

La représentation du temps et de l'espace ainsi que les modèles de raisonnements associés sont des thèmes largement étudiés en informatique, d'une manière générale, et en intelligence artificielle, en particulier. Ces thèmes sont de plus en plus importants dans de nombreux domaines de notre société, en particulier là où est disponible une très grande quantité d'informations et de services évoluant au cours du temps ou dans l'espace. Les techniques temporelles et/ou spatiales sont, par exemple, importantes dans : la gestion des grandes quantités de données, l'analyse et la fouille de ces données, la simulation et l'analyse de l'évolution temporelle de processus ; l'évaluation de la sécurité et la sûreté ; la gestion dynamique des connaissances ; la gestion de l'espace, la prévention des risques naturels, la modélisations des systèmes dynamiques et complexes, etc. Elles offrent une alternative ou un complément aux méthodes statistiques et mathématiques de modélisation de l'espace et du temps.

En complément des contributions dans les conférences généralistes dans les domaines de l'intelligence artificielle, des systèmes d'informations, des bases de données, du traitement automatique de la langue naturelle, des systèmes temps réel et de l'informatique théorique, l'atelier RTE a émergé comme l'atelier national qui réunit les chercheurs français s'intéressant à la dimension temporelle et/ou spatiale pour discuter des travaux en cours, projets et idées. L'atelier RTE s'est tenu pour la première fois en 2005 à Nice dans le cadre de la plateforme AFIA, a eu lieu ensuite à Nantes (semaine de la connaissance, 2006), à Grenoble (plateforme AFIA, 2007), à Montpellier (conférence SAGEO, 2008), et est cette année un atelier associé à la conférence RFIA qui est accueillie à Caen. Il est ouvert à la présentation de travaux de chercheurs et doctorants portant sur l'un ou plusieurs des thèmes suivants : modélisation du temps, modélisation de l'espace, raisonnement spatial qualitatif, raisonnement temporel qualitatif, traitement du langage naturel,

représentation de connaissances, fouille de données spatiales et temporelles, applications, etc. Pour cette cinquième manifestation, nous avons retenu sept soumissions qui présentent des travaux dans les différents champs cités. Ces présentations sont accompagnées d'un exposé de Jean-François Mari sur l'utilisation de modèles graphiques pour le raisonnement temporel et spatial.

Nous souhaitons que les participants à cet atelier trouvent matière à idées et discussions dans l'ensemble des présentations et que ces discussions puissent se prolonger lors de futures manifestations. Finalement, nous remercions chaleureusement les membres du comité de programme ainsi que les membres du comité d'organisation de RFIA, sans lesquels cet atelier n'aurait pu se tenir.

Maroua Bouzid
Florence Le Ber
Gérard Ligozat
Odile Papini

Comité de programme

Isabelle Bloch (ENST, Paris)
Maroua Bouzid (GREYC, Caen)
Jean-François Condotta (CRIL, Lens)
Mauro Gaio (LIUPPA, Pau)
Jérôme Gensel (LIG, Grenoble)
Christine Largouët (IRISA, Rennes)
Florence Le Ber (CEVH, Strasbourg / LORIA)
Gérard Ligozat (LIMSI, Orsay)
Philippe Muller (IRIT, Toulouse)
Odile Papini (LSIS, Marseille)
Pascal Poncelet (LGI2P, Nîmes)

Programme de l'atelier

- ◆ **9h10-10h Conférence invitée**

Jean-François Mari : quelques modèles graphiques pour le raisonnement temporel et spatial

- ◆ **10h-10h30 Session 1**

Adrien Delaye et Eric Anquetil : Apprentissage de relations spatiales floues pour la reconnaissance de tracés manuscrits

- ◆ **10h30-11h Pause-café**

- ◆ **11h-12h30 Session 1 (suite)**

Martin Cooper, Frédéric Maris, Pierre Régner : Complétude des algorithmes de planification temporellement expressifs

Philippe Balbiani, Olivier Gasquet, François Schwarzentruher : Knowledge in flatland

Florence Le Ber, Jean Lieber, Amedeo Napoli : Représentation temporelle qualitative de recettes de cuisine

- ◆ **12h30-14h Buffet**

- ◆ **14h-15h30 Session 2**

Laurent Etienne, Thomas Devogele, Alain Bouju : Analyse temps réel du comportement d'objets mobiles évoluant dans un espace ouvert

Thierry Le Pors, Thomas Devogele, Christine Chauvin : Représentation de l'espace et des déplacements dans la simulation de comportements humains – application au trafic maritime

Jean-François Mari, El Ghali Lazrak, Marc Benoît : Fouille de paysages agricoles – analyse des voisinages des successions d'occupation du sol

- ◆ **15h30-16h Pause Café**

- ◆ **15h30-17h30 Session 3 (Atelier Sources ouvertes et Services)**

Sommaire

- Adrien Delaye et Eric Anquetil : Apprentissage de relations spatiales floues pour la reconnaissance de tracés manuscrits*
... pages 9-12
- Martin Cooper, Frédéric Maris, Pierre Régnier : Complétude des algorithmes de planification temporellement expressifs*
... pages 13-19
- Philippe Balbiani, Olivier Gasquet, François Schwarzentruher : Knowledge in flatland*
... pages 21-24
- Florence Le Ber, Jean Lieber, Amedeo Napoli : Représentation temporelle qualitative de recettes de cuisine*
... pages 25-28
- Laurent Etienne, Thomas Devogele, Alain Bouju : Analyse temps réel du comportement d'objets mobiles évoluant dans un espace ouvert*
... pages 29-33
- Thierry Le Pors, Thomas Devogele, Christine Chauvin : Représentation de l'espace et des déplacements dans la simulation de comportements humains – application au trafic maritime*
... pages 35-37
- Jean-François Mari, El Ghali Lazrak, Marc Benoît : Fouille de paysages agricoles – analyse des voisinages des successions d'occupation du sol*
... pages 39-45

Apprentissage de relations spatiales floues pour la reconnaissance de tracés manuscrits

Adrien Delaye

Eric Anquetil

INSA de Rennes, Avenue des Buttes de Coësmes, F-35043 Rennes
UMR IRISA, Campus de Beaulieu, F-35042 Rennes
Université Européenne de Bretagne, France

{adrien.delaye, eric.anquetil}@irisa.fr

Résumé

Nous nous intéressons à la description et à la modélisation du positionnement relatif de tracés manuscrits à l'aide de la morphologie mathématique floue. Nous proposons une nouvelle approche pour l'apprentissage automatique de modèles de positionnement spatial basée sur des opérations de dilatation morphologiques. L'apport de ces nouveaux modèles pour la reconnaissance de formes est illustré par des expérimentations menées sur une base de gestes graphiques en-ligne.

Mots Clef

Morphologie mathématique floue, positionnement spatial, apprentissage, reconnaissance de tracés manuscrits.

1 Introduction

L'interprétation des relations spatiales joue un rôle important pour l'analyse de formes manuscrites composées, telle que des symboles complexes, des équations mathématiques, des schémas structurés... Pourtant, les techniques usuelles pour la description du positionnement relatif de tracés manuscrits reposent souvent sur des modèles simples, résumant chaque tracé à un seul point synthétique (méthode des centres de gravité), ou à son rectangle englobant, se contentant d'une représentation des relations spatiales grossière et peu conforme à l'intuition. Par ailleurs, il existe en analyse d'images des méthodes pour la description des relations spatiales qui prennent en compte les formes effectives des objets : c'est le cas de l'approche morphologique présentée par Bloch [1], également utilisée dans d'autres travaux [2, 3]. Le fait de tenir compte des formes effectives des objets à positionner constitue un avantage important pour la qualité de la description. En effet, la description est alors *adaptive*, et donc susceptible d'apporter une modélisation plus ro-

buste et précise du positionnement d'objets soumis à variabilité, comme c'est le cas pour les tracés manuscrits. Par son extension au cadre de la logique floue, la morphologie mathématique permet aussi de bien gérer l'ambiguïté du positionnement relatif de ces tracés sujets à une grande imprécision.

Dans ces travaux, nous proposons une méthode pour pouvoir apprendre, à partir d'exemples, des modèles *génératifs* décrivant une classe donnée de positionnement relatif, en s'appuyant sur le cadre de la morphologie mathématique floue. L'intérêt de tels modèles est de pouvoir guider un système de reconnaissance *structurel* par prédiction ou vérification de contraintes spatiales. Des résultats comparatifs obtenus sur une base de données de tracés manuscrits sont exposés.

2 Description de positionnement relatif par morphologie floue

Nous reprenons ici la démarche générale de la description du positionnement relatif d'objets telle que présentée dans [1]. Si la méthode originale est conçue pour traiter des objets flous dans des images, et peut être étendue à du positionnement d'objets en 3 dimensions, nous nous contentons d'en exposer les principes simplifiés à notre cadre applicatif : l'espace est réduit au plan, et nous ne considérons que des objets non flous. Le principe fondamental consiste à concevoir une relation spatiale comme un sous-ensemble flou défini dans le plan, dont la fonction d'appartenance associe à tout point du plan une valeur d'adéquation à la relation considérée par rapport à un objet de référence. Un exemple d'une telle relation est *être à droite de R*, où *R* est un objet de référence positionné dans l'espace de façon absolue. La vérification d'une relation de positionnement, pour un objet *A* par rapport à un objet *R* se fait en deux étapes.

La première étape consiste à définir la fonction modéli-

sant la zone de l'espace qui satisfait la relation spatiale par rapport à R . Intuitivement, ce *paysage flou* associe à un point du plan le degré avec lequel il vérifie la relation de positionnement décrite autour de la référence. Pour un objet R de référence donné, positionné dans le plan S , le paysage flou μ est construit en opérant une dilatation morphologique de R au moyen d'un élément structurant ν :

$$\forall P \in S, \mu(R)(P) = \max_{Q \in R} \nu(P - Q) \quad (1)$$

La forme de l'élément structurant ν détermine la relation spatiale considérée. En particulier, l'élément structurant doit être défini par une fonction d'appartenance floue pour permettre de construire un paysage flou. Par exemple, la forme suivante est choisie pour définir l'élément structurant modélisant une relation dans la direction α par rapport à R :

$$\forall P \in S, \nu_\alpha(P) = \max \left(0, 1 - \frac{2}{\pi} \arccos \frac{\overrightarrow{OP} \cdot \vec{u}_\alpha}{\|\overrightarrow{OP}\|} \right) \quad (2)$$

où O est le centre de l'élément structurant et u_α est le vecteur unitaire de direction α . La fonction d'appartenance de ν associe donc à un point du plan un degré entre 0 et 1, qui varie linéairement avec l'angle entre \overrightarrow{OP} et la direction α considérée. La figure 1 représente un tel élément structurant pour la relation à droite de, et la figure 2 illustre un exemple de paysage flou obtenu en dilatant une forme de référence par cet élément structurant. Les niveaux de gris représentent les valeurs des degrés d'appartenance des points du plan aux sous-ensembles flous : les points en blanc avec un degré de 1, les points en noir avec un degré nul.

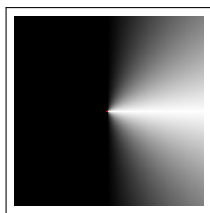


FIGURE 1 – Élément structurant ν_{droite}

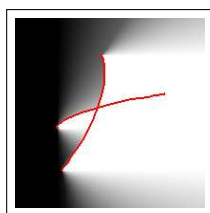


FIGURE 2 – Paysage flou être à droite de R

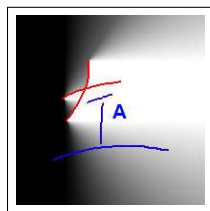


FIGURE 3 – Relation être à droite de R

Mesure	Valeur
$M^R(A)$	0,73
$N^R(A)$	0
$\Pi^R(A)$	1

TABLE 1 – Évaluation de la relation

La seconde étape consiste à évaluer avec quel degré l'objet A à positionner par rapport à R vérifie la relation spatiale considérée, en utilisant le paysage flou.

On peut pour cela considérer différents opérateurs qui agrègent les valeurs d'appartenance des points de A au paysage flou, comme la mesure du degré moyen (3), ou les mesures de nécessité (4) et de possibilité (5) [1] :

$$M^R(A) = \frac{1}{|A|} \sum_{P \in A} \mu(R)(P) \quad (3)$$

$$N^R(A) = \inf_{P \in A} \mu(R)(P) \quad (4)$$

$$\Pi^R(A) = \sup_{P \in A} \mu(R)(P) \quad (5)$$

La figure 3 reprend l'exemple de la figure 2 avec la représentation de l'objet A à positionner par rapport à la référence. Le tableau 1 donne, pour le couple d'objets de la figure 3, les résultats obtenus par les trois mesures.

3 Apprentissage de modèles de positionnement

L'utilisation de la morphologie mathématique permet de prendre en compte la forme de l'objet de référence dans la définition du paysage flou, et la description s'adapte donc naturellement à ses spécificités. De plus, de par sa définition floue, la relation spatiale ainsi construite admet une certaine flexibilité précieuse pour gérer l'imprécision qui intervient dans le positionnement relatif des tracés manuscrits. L'objectif de nos travaux est, en s'appuyant sur le cadre morphologique, d'apprendre automatiquement des modèles de positionnement *génératifs*, c'est-à-dire permettant de répondre à deux types de questions :

1. étant donné un objet de référence et ce modèle de positionnement, dans quelle zone du plan peut/doit se trouver l'objet à positionner ?
2. étant donnés deux objets, avec quel degré respectent-ils ce modèle de positionnement ?

Nous proposons de considérer les 4 directions principales une à une, et d'apprendre pour chacune un élément structurant spécifique au modèle de positionnement considéré. Le but est de pouvoir déterminer, selon chaque direction, un paysage flou décrivant la zone du plan admise pour le modèle. La construction du modèle global est faite ultérieurement par fusion de ces paysages flous directionnels.

3.1 Apprentissage des éléments structurants

À partir d'un grand nombre d'exemples de couples d'objets (R_i, A_i) , instances d'un modèle de positionnement que l'on veut apprendre, nous voulons construire les sous-ensembles flous décrivant les degrés atteints par les points d'apprentissage selon les quatre relations directionnelles *haut*, *bas*, *gauche* et *droite*.

Nous choisissons pour cela d'estimer, pour chaque direction, un sous-ensemble flou décrivant la distribution

des degrés de conformité des points des objets A_i par rapport à leur objet de référence R_i . Cette approximation est fournie par des prototypes flous obtenus par un algorithme de clustering possibiliste [4].

Notons Φ le sous-ensemble flou qui modélise les degrés admis par les points d'apprentissage selon la relation directionnelle droite, alors $(\Phi \circ \nu_{droite})$ définit un nouvel élément structurant. La figure 4(a) représente un tel élément structurant, qui décrit le modèle de positionnement des objets R et A de la figure 3 selon la direction droite. Le paysage flou $\mu_{droite}^{\Phi}(R)$, résultat de la dilatation d'un objet de référence R par cet élément structurant, attribue aux points du plan leur degré de conformité avec la relation apprise, toujours selon la direction *droite*. Ainsi, le paysage flou de la figure 4(b) représente la relation *être à droite de R dans la même mesure que A d'après le modèle appris*. Les points les plus clairs sont donc ceux pour lesquels les degrés de validité de la relation *être à droite de R* sont les plus conformes aux degrés mesurés sur les points d'apprentissage.

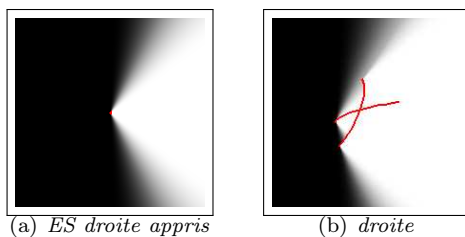


FIGURE 4 – Élément structurant appris (a) et paysages directionnel correspondant (b) pour la direction *droite*

3.2 Fusion par intersection floue

Le procédé décrit dans la partie précédente est répété sur chacun des axes considérés : les 4 directions *haut*, *bas*, *gauche*, *droite*, ainsi qu'une dimension supplémentaire de distance (pour plus de détails sur la modélisation de la distance par élément structurant flou, le lecteur intéressé peut se référer à [5]). Le modèle global est obtenu par intersection des paysages flous obtenus pour chacun des axes, traduisant le fait qu'un point est bien positionné par rapport à la référence selon un modèle s'il est dans la région d'acceptabilité apprise pour chacun des axes. Nous choisissons l'opérateur *produit* pour réaliser cette intersection, mais tout opérateur de conjonction floue (T-norme) peut être envisagé.

La figure 5 présente les paysages flous de directions (a, b, d) et de distance (c) décrivant les régions d'acceptabilité pour un objet de référence, ainsi que le modèle global obtenu par intersection (e). Ce paysage flou global, $\mu^{\Phi}(R)$ modélise la zone du plan admise pour un objet par rapport à R selon le modèle appris (cf. problème 1 évoqué en préambule de la section 3), et il peut être exploité pour évaluer le positionnement d'un objet

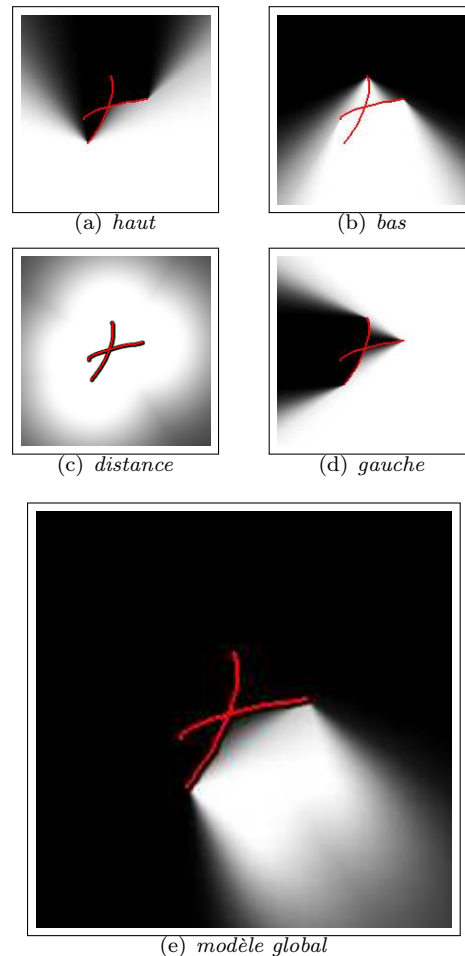


FIGURE 5 – Paysages appris pour les 3 autres directions (a,b,d) ainsi que la distance (c), et paysage global obtenu par intersection floue (e)

A (cf. problème 2 évoqué en préambule de la section 3).

4 Expérimentations

Nous avons comparé les performances de la modélisation du positionnement proposée ici sur une tâche de reconnaissance de gestes manuscrits. Les données expérimentales regroupent un ensemble de 18 classes de gestes graphiques correspondant à des gestes d'édition (suppression d'un caractère, espace, retour chariot...) ou bien à la saisie de signes diacritiques (accents, cédille, apostrophe) ou encore de ponctuation (plus de détails dans [5]). Ces gestes sont saisis en contexte, c'est-à-dire qu'ils sont associés à une lettre de référence (lettre éditée, lettre support du signe diacritique ou précédant le symbole de ponctuation). Puisque plusieurs classes de gestes ont une forme très semblable (par exemple la virgule et l'apostrophe), la modélisation de leur positionnement relativement à la référence est nécessaire pour une bonne reconnaissance.

Un premier test de reconnaissance est réalisé, où les

gestes sont décrits par des caractéristiques de formes, ainsi que deux mesures de leurs dimensions relatives par rapport aux références associées. Le positionnement est décrit par une mesure de distance et 4 mesures des degrés moyens de positionnement flou par rapport aux quatre directions principales, d'après l'idée exposée dans de précédents travaux [6]. Le taux de reconnaissance obtenu au moyen d'un classifieur SVM dans cette expérimentation est de 96,03%, attestant de la bonne qualité de la description du positionnement relatif par ce premier jeu de caractéristiques. Pour chacune des 18 classes, nous apprenons dans un second temps un modèle de positionnement d'après la méthode exposée à la section 3. La figure 6 représente les modèles appris pour 4 classes, déployés sur des lettres en référence. À partir de ces modèles nous élaborons 18 caractéristiques de description du positionnement : pour un geste à reconnaître, nous mesurons le degré d'adéquation de son positionnement avec les 18 modèles. Le jeu de caractéristiques est complété par les deux mesures des dimensions relatives du geste par rapport à sa référence, et par les mêmes caractéristiques de description de forme que dans le premier test. Le taux de reconnaissance obtenu dans ce second test s'élève à 96,16 %, ce qui constitue une réduction de l'erreur de l'ordre de 3%.

Ces résultats légèrement meilleurs que ceux obtenus par utilisation directe des opérateurs de morphologie floue attestent de la qualité des modèles génératifs présentés. De plus, ces modèles de positionnement semblent pertinents pour permettre un usage prédictif comme le confirment visuellement les exemples de la figure 6. On peut envisager par exemple que pour guider l'analyse des éléments dans une application de reconnaissance d'équations mathématiques, il soit judicieux de limiter spatialement l'analyse aux zones admissibles pour les relations considérées : *en indice*, *à droite*, *en exposant*. . . Cet usage ne pouvait pas être envisagé avec l'utilisation directe des opérateurs de morphologie mathématique tels qu'ils ont été exploités jusqu'à présent.

5 Conclusion

Nous avons formalisé de nouveaux modèles de relations spatiales, introduit une méthode pour leur apprentissage automatique et montré leur efficacité sur une base de données de gestes manuscrits. La force de ces modèles réside dans leur espace de représentation qui est défini lui-même par rapport à l'objet de référence grâce à la morphologie mathématique. Cela leur confère une propriété d'adaptativité aux formes des objets considérés. Un procédé pour l'apprentissage de ces modèles à partir de données permet de garantir une certaine finesse et une qualité de la modélisation. Enfin, l'intérêt majeur de ce nouveau type de modèles de positionnement est leur aspect génératif qui permet d'en faire usage pour prédire et vérifier des contraintes

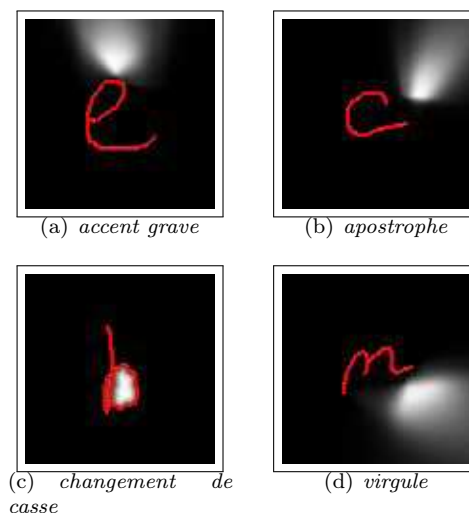


FIGURE 6 – Représentation des modèles appris pour les classes “accent grave”, “apostrophe”, “changement de casse” et “virgule”

de positionnement, notamment dans le cas d'une analyse structurale de formes manuscrites.

Références

- [1] I. Bloch. Fuzzy relative position between objects in image processing : a morphological approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(7) :657–664, 1999.
- [2] P.D. Gader. Fuzzy spatial relations based on fuzzy morphology. In *Proc. Sixth IEEE International Conference on Fuzzy Systems*, volume 2, pages 1179–1183 vol.2, 1997.
- [3] R.C. Cinbis and S. Aksoy. Relative position-based spatial relationships using mathematical morphology. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 97–100. IEEE Computer Society, 2007.
- [4] R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering. *IEEE transactions on fuzzy systems*, 1(2) :98–110, 1993.
- [5] A. Delaye and E. Anquetil. Modélisation du positionnement relatif de tracés manuscrits par morphologie mathématique floue. In *Actes du 17ème Congrès Francophone de Reconnaissance des Formes et d'Intelligence Artificielle*, 2010.
- [6] F. Bouteruche, S. Macé, and E. Anquetil. Fuzzy relative positioning for on-line handwritten stroke analysis. In *Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition*, pages 391–396, 2006.

Complétude des algorithmes de planification temporellement expressifs

Martin Cooper

Frédéric Maris

Pierre Régnier

IRIT, Université Paul Sabatier,
118 route de Narbonne
31062 Toulouse, cedex 9, France.
{cooper, maris, regnier}@irit.fr

Résumé

Un des challenges actuels de la planification est la prise en compte de la dimension temporelle qui permet de mieux modéliser des problèmes issus du monde réel. Parmi ceux-ci, nombreux sont ceux dont la résolution nécessite la concurrence de certaines actions (problèmes temporellement expressifs). Cependant, les planificateurs qui peuvent résoudre ce type de problèmes sont pour la plupart incomplets car ils s'avèrent incapables de trouver une solution à une catégorie de problèmes comportant des ensembles cycliques d'actions (problèmes temporellement cycliques). Dans cet article, nous caractérisons les langages temporels qui permettent de représenter ces problèmes temporellement cycliques, puis nous proposons une méthode de transformation quadratique de ces problèmes en des problèmes acycliques équivalents. L'application de notre algorithme permet de rendre complets les planificateurs précédents.

Mots Clefs

Planification, temps, problèmes temporellement expressifs, problèmes temporellement cycliques.

Abstract

In order to correctly model certain real-world planning problems, it is essential to take into account time. This is the case for problems requiring the concurrent execution of actions (known as temporally-expressive problems). However, we show in this paper that most existing planners which solve this type of problem are, in fact, incomplete. They cannot guarantee to find a solution to a problem involving sets of cyclically-dependent actions (which we call temporally-cyclic problems). We characterize those temporal planning languages which can express temporally-cyclic problems. We also present a polynomial-time algorithm which transforms a temporally-cyclic problem into an equivalent acyclic problem. Applying our transformation restores the completeness of existing temporal planners.

Keywords

Planning, time, temporally-expressive problems, temporally-cyclic problems.

1 Introduction

Pour envisager la résolution de problèmes réels, l'un des challenges actuels de la planification est la prise en compte du temps. De plus, un grand nombre de problèmes réels, pour être résolus ou exécutés de manière plus efficace, nécessitent la prise en compte d'actions duratives concurrentes. Dans cet article nous nous intéresserons spécifiquement à la résolution de problèmes temporellement expressifs, c'est-à-dire de problèmes que l'on ne peut résoudre qu'en faisant intervenir la concurrence des actions. Gérer une aérogare, une gare, cuire des céramiques, faire la cuisine, planter un clou... sont quelques exemples de tels problèmes.

La majeure partie des planificateurs temporels actuels s'avère incapable de résoudre des problèmes de ce type [6]. Si l'on excepte les planificateurs de type HTN comme IxTeT [9], [13] ou HSTS [19], seuls les systèmes CRIKEY3 [5], VHPOP [28], LPGP [14], TLP-GP [15] [16], TM-LPSAT [26], [11] et STEP [12] résolvent ce type de problèmes.

Malheureusement ces planificateurs sont pour la plupart incomplets car ils s'avèrent incapables de trouver une solution aux problèmes qui comportent des ensembles cycliques d'actions (problèmes temporellement cycliques). Un exemple simple de problème de ce type est l'intégration, en temps contraint, de deux logiciels dont la réalisation serait confiée à des sous-traitants différents, chacun ayant besoin de connaître les spécifications de l'autre logiciel pour pouvoir les interfacer.

Dans cet article, après avoir résumé les principaux travaux qui traitent de cette problématique (section 2) puis fixé notre cadre de travail (section 3), nous caractérisons les sous-langages temporels issus du langage PDDL2.1. (Planning Domain Description Language) [18] [8] et qui permettent de représenter les problèmes temporellement cycliques (section 4). Nous proposons ensuite une méthode de transformation quadratique de ces problèmes en des problèmes acycliques équivalents (section 5). L'utilisation de notre algorithme permet de rendre complets les planificateurs précédents. Nous terminons en exposant les perspectives ouvertes par notre travail.

2 Travaux connexes

Les planificateurs temporellement expressifs sont essentiellement basés sur trois types d'algorithmes [17] :

La recherche dans les espaces d'états :

[6] propose TEMPO, un algorithme temporellement expressif basé sur la recherche dans les espaces d'états étendus (au temps) dans lequel les décisions du type « quand exécuter une action » sont reportées jusqu'à ce que toutes les décisions du type « quelle action exécuter » aient été prises (lifting over time). Le planificateur CRIKEY3 [10], [5] planifie de manière classique et, en cas d'échec, bascule vers un algorithme de type TEMPO.

La recherche dans les espaces de plans partiels :

Les planificateurs dans les espaces de plans partiels (POP) ont été étendus avec succès au cadre temporellement expressif avec les planificateurs VHPOP [28] et DT-POP [23]. Mais de nombreux autres planificateurs, anciens pour la plupart, travaillent dans les espaces de plans hiérarchiques (HTN). Ils utilisent une logique temporelle d'instant ou d'intervalles et un module spécifique (Time Map Manager) qui gère les contraintes temporelles. C'est le cas de planificateurs comme FORBIN [7], HSTS [19], IXTET [9], [13], TEST [20], TIMELOGIC [2], TLP [25], TRIPTIC [22]. Leurs langages de représentation sont très expressifs mais présentent de nombreuses différences par rapport à PDDL2.1, ce qui les rend très difficiles à comparer avec les autres systèmes. Les quelques tests qui ont pu être réalisés montrent des performances limitées.

Les extensions temporelles de GRAPHPLAN :

L'utilisation des graphes de planification [3] a aussi été étendue aux problèmes temporellement expressifs. Dans LPGP [14] et TM-LPSAT [26] les actions duratives sont décomposées en trois actions instantanées (start, invariant, end). Ces planificateurs utilisent tous deux un solveur pour l'extraction d'une solution. Cependant, si LPGP fonctionne en recherche arrière dans le graphe de planification tout en maintenant la consistance d'une base de contraintes temporelles, TM-LPSAT code entièrement le graphe et les contraintes temporelles puis fait appel au solveur LPSAT [27]. Une méthode comparable à celle de TM-LPSAT est également décrite dans [11]. Les actions sont décomposées en deux actions simples puis le problème est codé en CSP selon une sémantique basée sur des opérateurs modaux et qui englobe le codage du graphe de planification. Le planificateur TLP-GP [15] [16] utilise des méthodes similaires à celles de LPGP et TM-LPSAT. Il délègue une part plus importante du travail de recherche au solveur et montre de bonnes performances par rapport à LPGP, VHPOP2.2 et CRIKEY3.

3 Les problèmes temporellement cycliques

Dans cet article, nous nous plaçons dans le cadre du langage PDDL2.1 et de ses sous-langages. Si l'on autorise le recouvrement de plusieurs instances d'une même action, la complexité de l'existence d'un plan-solution devient EXSPACE-complète [21]. Dans ce papier, nous nous limitons donc, comme dans la quasi-totalité des travaux qui ont été réalisés en planification temporelle, à n'autoriser qu'une seule instance d'une même action en parallèle, ce qui permet de conserver la complexité de la planification classique qui est PSPACE-complète [4]. De plus, deux instances d'une même actions présentes dans un même plan devront ne pas être contiguës au sens de la primitive "meets" de Allen [1].

Nous nous intéressons ici aux seuls planificateurs capables de résoudre des problèmes temporellement expressifs [6]. Nous allons commencer par montrer que la plupart d'entre eux ne sont pas complets car ils s'avèrent incapables de trouver une solution à un certain type de problème que nous allons maintenant définir. Prenons l'exemple simple du problème d'interfaçage de deux logiciels que nous avons introduit en section 1. Un plan-solution à ce petit problème temporellement expressif est donné figure 1. Chaque rectangle représente une action durative $x = (\langle \text{conditions} \rangle, \langle \text{effets} \rangle, \langle \text{durée} \rangle)$. Les conditions $\text{Cond}(x)$ apparaissent au-dessus de l'action et les effets $\text{Eff}(x)$ au-dessous. Dans ce problème, l'état initial est vide, le but consiste à obtenir les deux logiciels interfacés (effets r et s) et :

- l'action A (de durée 4) correspond à la réalisation du premier logiciel pour laquelle on a besoin de connaître, afin de la mener à son terme, les spécifications du deuxième logiciel (condition p).
- l'action B (de durée 2) correspond à la réalisation concurrente du deuxième logiciel pour laquelle on a besoin de connaître, afin de la terminer, les spécifications du premier logiciel (condition q).

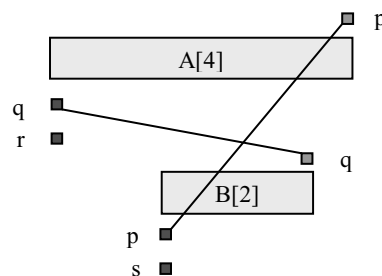


Fig. 1 – Plan-solution au problème de la section 1

Ce problème est caractéristique d'une classe de problèmes que nous appellerons "problèmes temporellement cycliques" et que l'on peut formaliser ainsi :

Définition 1 (ensemble cyclique d'actions) : Un ensemble d'actions $Q = \{a_1, \dots, a_n\}$ est un ensemble cyclique s'il existe un ensemble de propositions $\{p_1, \dots, p_n\}$ telles que $\forall i \in \{1, \dots, n-1\} p_i \in \text{Eff}(a_i) \cap \text{Cond}(a_{i+1})$ et $p_n \in \text{Eff}(a_n) \cap \text{Cond}(a_1)$. On notera cet ensemble cyclique ainsi : $a_1 \rightarrow p_1 \rightarrow a_2 \dots \rightarrow p_n \rightarrow a_1$.

Définition 2 (problème temporellement cyclique) : Soit un problème de planification $\langle \mathcal{A}, I, \mathcal{B} \rangle$ dans lequel \mathcal{A} est l'ensemble des actions du problème, I son état initial et \mathcal{B} le but à atteindre. Ce problème est un problème temporellement cyclique ssi \mathcal{A} contient un ensemble cyclique d'actions.

La plupart des planificateurs temporellement expressifs de l'état de l'art comme CRIKEY3, LPGP ou TLP-GP s'avèrent incapables de produire le plan-solution de la figure 1 alors que VHPOP le trouve. CRIKEY3 utilise un algorithme de recherche dans les espaces d'états étendus alors que LPGP et TLP-GP construisent des graphes de planification avant d'effectuer une recherche en arrière. Leur point commun est que, soit la recherche de solution, soit la construction du graphe de planification s'effectue en avant à partir de l'état initial.

En utilisant une recherche avant guidée par l'état initial ou la construction d'un graphe de planification, il n'est pas possible de produire $r \in \mathcal{B}$ ou $s \in \mathcal{B}$ car aucune des conditions $\{p, q\}$ nécessaires à la sélection des actions $\{A, B\}$ dans le plan ne peut être produite à partir de I .

En recherche arrière, cette difficulté est levée car, comme le choix des actions est guidé par les buts, les actions A et B seront obligatoirement sélectionnées. Par contre, cela ne suffit pas à garantir la complétude de l'algorithme de recherche. En effet, si l'heuristique de choix des actions ne privilégie pas les actions déjà présentes dans le plan pour établir les sous-buts, l'algorithme peut boucler indéfiniment. Ainsi, on pourrait être amené à insérer une nouvelle instance de A pour produire q pour la dernière instance de B sélectionnée (au lieu d'utiliser l'instance de A déjà présente dans le plan). Puis, pour établir p pour cette dernière instance de A, on pourrait insérer une nouvelle instance de B, etc.

4 Expressivité temporelle et ensembles cycliques d'actions

Les travaux décrits dans [6] introduisent la notation $L_{\text{effets}}^{\text{conditions}}$ pour représenter l'expressivité d'un langage de planification temporel, où $\langle \text{conditions} \rangle$ et $\langle \text{effets} \rangle$ représentent les instants auxquels les conditions doivent être vraies et ceux pendant lesquels les effets peuvent se produire. Les valeurs prises par $\langle \text{conditions} \rangle$ et $\langle \text{effets} \rangle$ peuvent être les suivantes :

- s : instant de démarrage d'une action (*at start*) ;

- e : instant de fin d'une action (*at end*) ;
- o (ne concerne que les conditions) : porte sur la durée totale de l'action (*over all*).

Avec cette notation, $L_{s,e}^o$ désigne par exemple un langage pour lequel les conditions d'une action doivent être vraies durant toute l'exécution, et les effets doivent survenir à l'instant de début ou à l'instant de fin d'une instance de l'action. PDDL2.1 est ainsi un langage temporellement expressif $L_{s,e}^{s,o,e}$. Ils proposent également une méthode permettant de prouver qu'un domaine est temporellement expressif. Ceci les amène à diviser l'espace des langages temporels en deux ensembles :

- Les langages temporellement expressifs permettant de représenter des problèmes pour la résolution desquels la simultanéité de certaines actions est requise ;
- Les langages temporellement simples qui ne permettent pas.

Cushing et al. [6] proposent également les définitions suivantes et ils démontrent le théorème d'expressivité temporelle que nous donnons ci-après.

Définition 3 (condition-avant/condition-après) :

- Une *condition-avant* est une condition qui est nécessaire au moment, ou avant, qu'au moins l'un des effets de l'action soit produit.
- Une *condition-après* est une condition qui est nécessaire au moment, ou après, qu'au moins l'un des effets de l'action soit produit.

Définition 4 (trou temporel) :

- Un trou entre deux intervalles temporels disjoints et qui ne se rencontrent pas (au sens de la primitive "meets" de Allen [1]) est l'intervalle qui se trouve entre eux (et tel que l'union de ces trois intervalles forme un intervalle unique).
- Une action a un *trou temporel* s'il existe un trou entre une *condition-avant* et un effet, ou un trou entre un effet et une *condition-après*, ou entre deux effets.

Théorème de [6] : Un sous-langage de PDDL2.1 est temporellement simple si et seulement si il interdit les trous temporels.

Nous allons maintenant montrer qu'un certain type de trou temporel permet à un langage d'exprimer des problèmes temporellement cycliques.

Notations : Soit une action a , avec $c \in \text{Cond}(a)$ et $e \in \text{Eff}(a)$. Dans un plan temporel, $\tau(c \rightarrow a)$ représentera l'instant où a nécessite c et $\tau(a \rightarrow e)$ représentera l'instant où a produit e .

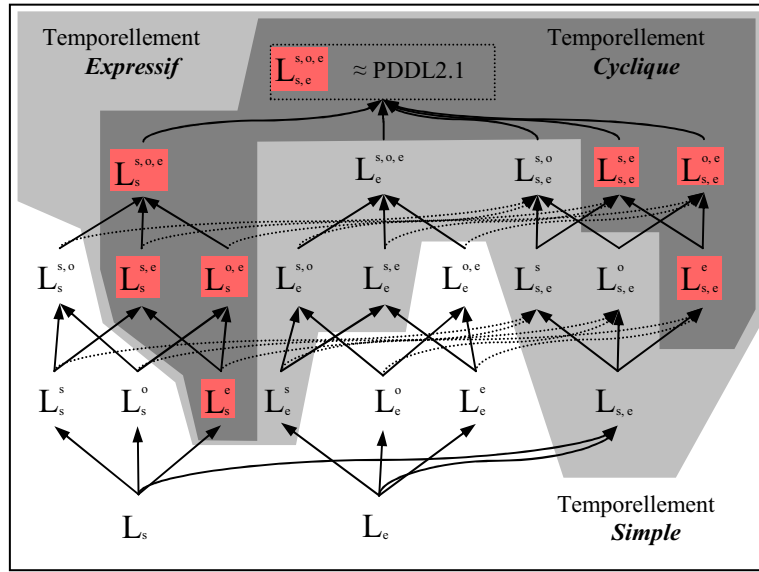


Fig. 2 – Taxinomie des langages temporels et leur expressivité

Définition 5 (langage temporellement cyclique/acyclique) : Un langage sera dit temporellement cyclique ssi il autorise l'existence d'ensembles cycliques d'actions. Dans le cas contraire, ce langage sera dit temporellement acyclique.

Théorème 1 : Un sous-langage de PDDL2.1 est temporellement cyclique ssi il autorise les trous temporels entre un effet et une *condition-après*.

Ce théorème, dont la preuve est donnée ci-après, permet d'affiner la taxinomie des langages temporels proposée par [6] en distinguant une sous-classe des langages temporellement expressifs correspondant aux langages temporellement cycliques. Cette nouvelle taxinomie est donnée à la figure 2 ci-dessus.

Preuve du théorème 1 : (1) Montrons d'abord que si un sous-langage de PDDL2.1 autorise les trous temporels entre un effet et une *condition-après* alors il autorise les ensembles cycliques d'actions. Le problème résolu figure 1 comporte deux actions A et B et montre que L_s^e autorise les ensembles cycliques d'actions ($A \rightarrow q \rightarrow B \rightarrow p \rightarrow A$). Comme tous les sous-langages de PDDL2.1 qui autorisent les trous temporels entre un effet et une *condition-après* sont des sur-langages de L_s^e [6], tous ces sous-langages autorisent les ensembles cycliques d'actions. (2) Montrons maintenant que si un sous-langage de PDDL2.1 autorise les ensembles cycliques d'actions alors il autorise les trous temporels entre un effet et une *condition-après*. Soit $\langle \mathcal{A}, I, \mathcal{B} \rangle$ un problème de planification temporel. Il existe un ensemble d'actions $Q = \{a_1, \dots, a_n\} \subseteq \mathcal{A}$ et un ensemble de propositions $\{p_1, \dots, p_n\}$ tels que $(\forall i \in \{1, \dots, n-1\}) p_i \in \text{Eff}(a_i) \cap \text{Cond}(a_{i+1}) \wedge$

$(p_n \in \text{Eff}(a_n) \cap \text{Cond}(a_1))$ (cycle $a_1 \rightarrow p_1 \rightarrow a_2 \rightarrow p_2 \rightarrow a_3 \rightarrow p_3 \dots \rightarrow a_n \rightarrow p_n \rightarrow a_1$). D'où $(\forall i \in \{1, \dots, n-1\}) \tau(a_i \rightarrow p_i) < \tau(p_i \rightarrow a_{i+1}) \wedge (\tau(a_n \rightarrow p_n) < \tau(p_n \rightarrow a_1))$. Supposons qu'il ne puisse exister de trou temporel entre un effet et une *condition-après* d'une action. Alors, $(\forall i \in \{1, \dots, n-1\}) \tau(p_i \rightarrow a_{i+1}) \leq \tau(a_{i+1} \rightarrow p_{i+1}) \wedge (\tau(p_n \rightarrow a_1) \leq \tau(a_1 \rightarrow p_1))$. Donc $\tau(a_1 \rightarrow p_1) < \tau(p_1 \rightarrow a_2) \leq \tau(a_2 \rightarrow p_2) < \tau(p_2 \rightarrow a_3) \leq \dots \leq \tau(a_n \rightarrow p_n) < \tau(p_n \rightarrow a_1) \leq \tau(a_1 \rightarrow p_1)$. D'où $\tau(a_1 \rightarrow p_1) < \tau(a_1 \rightarrow p_1)$, ce qui est impossible. Il existe donc un trou temporel entre un effet et une *condition-après*.

Nous allons maintenant montrer qu'il est possible de réduire PDDL2.1 à $L_{s,e}^{s,o}$ en modélisant de façon différente les actions qui possèdent au moins un trou temporel entre un effet et une *condition-après*.

5 Transformation d'un problème temporellement cyclique en problème temporellement acyclique

Nous allons ici montrer comment passer d'un problème comportant des cycles potentiels exprimé dans un langage $L_{s,e}^{s,o,e}$, à un problème équivalent exprimé dans un langage $L_{s,e}^{s,o}$ dont on a montré - cf. théorème 1 - qu'il n'autorisait pas les cycles. On pourra donc utiliser la recherche avant pour résoudre l'ensemble des problèmes temporellement expressifs. En pratique, selon les problèmes, on pourra transformer toutes les actions en se ramenant à un langage $L_{s,e}^{s,o}$ ou ne transformer que celles qui peuvent entraîner la formation de cycles en restant ainsi dans un langage $L_{s,e}^{s,o,e}$.

La preuve du Théorème 1 montre qu'une condition nécessaire pour qu'un problème ait un plan-solution temporellement cyclique est l'existence d'un ensemble cyclique d'actions $a_1 \rightarrow p_1 \rightarrow a_2 \dots \rightarrow p_n \rightarrow a_1$ impliquant un trou temporel entre un effet p_i et une *condition-après* p_{i-1} d'une action a_i .

Pour détecter de telles actions, nous construisons le graphe de causalité $G = \langle V, E \rangle$ où l'ensemble des sommets V est l'union de l'ensemble des actions et de l'ensemble des propositions, et l'ensemble des arcs orientés $E = \{ \langle p, a \rangle : p \in \text{Cond}(a) \} \cup \{ \langle a, p \rangle : p \in \text{Eff}(a) \}$. Pour chaque action candidate A (i.e. une action qui présente un trou entre un effet et une *condition-après* p), nous testons une condition nécessaire à l'existence d'un cycle : le lien $p \rightarrow A$ fait-il partie d'un cycle dans le graphe de causalité. Si ce n'est pas le cas, alors il n'est pas nécessaire de transformer A .

De manière générale, toute action A qui nécessite des *conditions-après* pouvant causer un cycle doit être transformée en une action A' pour laquelle ces conditions ont été supprimées (pour casser le cycle). Chaque condition p de ce type va être reformulée comme un lien entre A et l'action B susceptible d'être la dernière, dans l'ordre temporel, à pouvoir produire cette condition pour A . Ce lien devra être actif à partir du moment où A sera présente dans le plan. Cette activation de lien sera vérifiée par une action $LC_{p,A}$ (Link Check de p par B pour A).

Pour transformer une action A afin de supprimer une *condition-après* p , on doit :

- Ajouter un prédicat $\text{Linked}(p,A)$ à l'état initial et au but du problème. Ce prédicat de lien permettra, quand

A est activée - et qu'elle détruit $\text{Linked}(p,A)$, cf. ci-après - de forcer l'apparition de l'action $LC_{p,A}$ pour rétablir $\text{Linked}(p,A)$ (cf. ci-après).

- Transformer A en A' :
 - en supprimant la *condition-après* p ,
 - en ajoutant un prédicat $\text{LC-Active}(p,A)$ à l'instant où p était nécessaire pour dénoter le fait que p doit être produite par une action avant sa consommation par A ,
 - en détruisant le prédicat $\text{Linked}(p,A)$, qui était présent dans l'état initial, ce qui va forcer (ce prédicat étant présent dans le but) le déclenchement de l'action $LC_{p,A}$ pour le rétablir.
- Transformer toute action B qui produit a en B' en détruisant le prédicat $\text{LC-Active}(p,A)$ à l'instant où B produit p pour dénoter le fait que p doit être produit (par au moins une des actions B) avant sa consommation par A .
- Rajouter une action instantanée $LC_{p,A}$ (Link Check) qui a pour conditions p et $\text{LC-Active}(p,A)$ et pour effet $\text{Linked}(p,A)$.

Ces nouveaux prédicats et actions ne peuvent introduire de nouveaux trous temporels entre effets et *conditions-après* puisque toutes les conditions ajoutées lors de la transformation, le sont au début des actions. Si nous reprenons notre exemple, pour une action A , la transformation se fera ainsi :

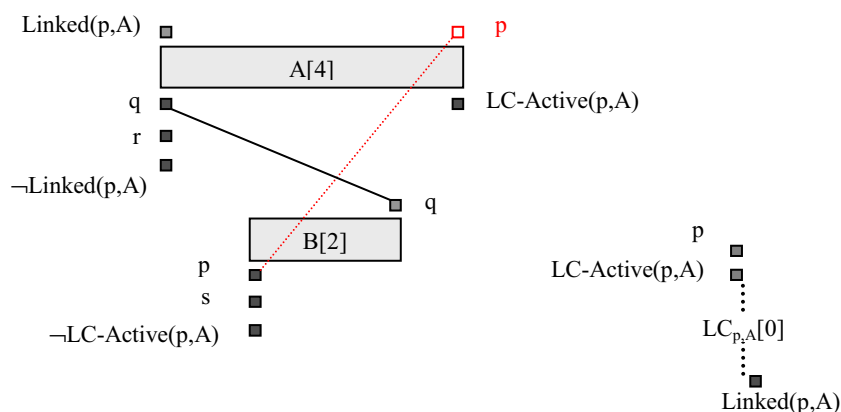


Fig. 3 – Exemple de transformation d'un problème de $L_{s,e}^{s,o,e}$ dans $L_{s,e}^{s,o}$.

Cette représentation du problème est équivalente au problème original et la transformation inverse d'une planification est réalisée en supprimant les actions $LC_{p,A}$.

Algorithme Transformation

```

Candidates ← {actions ayant un trou effet /
              condition-après}
Tant que Candidates ≠ ∅
  Retirer A de Candidates ;
  Pour chacune des conditions-après p de A
    générant un trou
    Générer le graphe de causalité G à
      partir de p → A ;
    Si p → A fait partie d'un cycle dans G
      alors
        Transformer_Cond(A,p,A') ;
        Pour toute action B qui produit p
          Transformer_Eff(B,p,B') ;
        Fin Pour ;
        Ajouter l'action  $LC_{p,A}$  au problème ;
      Fin Si ;
    Fin pour ;
  Fin Tant que ;
Fin Transformation.

```

Fig. 4 – Algorithme de transformation utilisant le graphe de causalité

La transformation totale d'un problème dans un langage temporellement acyclique $L_{s,e}^{s,o}$ peut être réalisée par la suppression systématique de toutes les conditions-après (sans tester la présence de cycles dans le graphe de causalité). Notons n la taille du problème, i.e. $n = \sum(|\text{Eff}(a)| + |\text{Cond}(a)|)$ où la somme porte sur toutes les actions a (en supposant, sans perte de généralité, que chaque action possède au moins une condition ou un effet). Alors cet algorithme simple a une complexité de $O(n^2)$. En effet, pour chaque action A avec une *condition-après* $p \in \text{Cond}(A)$, nous pouvons avoir à transformer toutes les autres actions B . La transformation des actions A et B , pour une *condition-après* donnée $p \in \text{Cond}(A)$, nécessite un temps constant.

Dans le pire des cas, la complexité de la transformation partielle d'un problème, dans lequel nous éliminons seulement les conditions-après qui peuvent mener à des cycles (détectées par l'analyse du graphe de causalité) est également $O(n^2)$, mais cet algorithme présente l'avantage d'introduire un nombre généralement moins important de nouvelles actions. L'algorithme est donné Fig. 4. Pour chaque action A et chaque *condition-après* $p \in \text{Cond}(A)$, nous commençons par tester s'il existe un ensemble

cyclique d'actions contenant $p \rightarrow A$. Ceci peut être réalisé en trouvant les composantes fortement connexes du graphe de causalité G en utilisant l'algorithme de [24]. Sur un graphe $\langle V, E \rangle$, cet algorithme a une complexité en temps de $O(|V| + |E|)$. Mais c'est en fait $O(n)$ puisque n est exactement égal au nombre d'arcs $|E|$. Une transformation complète peut donc être réalisée en $O(n^2)$ d'après l'argument du paragraphe précédent.

Dans les deux cas, le nombre de nouvelles actions $LC_{p,A}$ (ainsi que le nombre de nouveaux prédicats $LC\text{-Active}(p,A)$ et $\text{Linked}(p,A)$) est borné par le nombre de *conditions-après* présentes dans les actions du problème. La taille du problème transformé est donc linéaire par rapport à la taille du problème original.

6 Conclusion et perspectives

Dans cet article, nous avons montré que la plupart des planificateurs temporellement expressifs ne sont pas complets pour tous les sous-langages de PDDL2.1. Beaucoup sont en particulier incapables de résoudre des problèmes comportant des ensembles cycliques d'actions. Nous avons proposé une méthode qui permet, grâce à une transformation polynomiale des problèmes comportant de tels cycles, de rendre l'ensemble de ces planificateurs complets.

Les perspectives ouvertes par ce travail portent maintenant sur la complétude d'algorithmes utilisant des langages encore plus expressifs. Nous travaillons sur l'extension des résultats présentés dans ce papier à des problèmes dont les actions sont définies en utilisant des intervalles temporels quelconques auxquels il est possible d'associer différentes modalités. Ces modalités peuvent exprimer la persistance, une certaine forme d'incertitude, de choix ou de transitions continues.

Bibliographie

- [1] Allen J.F. (1984). Towards a General Theory of Action and Time. In *Artificial Intelligence 23(2)*, p. 123-154, 1984.
- [2] Allen J.F. & Koomen J.A.G.M. (1983). Planning Using a Temporal World Model. In *Proc. of 8th International Joint Conference on Artificial Intelligence*, pp. 741-747, 1983.
- [3] Blum A.L. & Furst M.L., (1995). Fast planning through planning-graphs analysis. In *Proc. of 14th International Joint Conference on Artificial Intelligence*, pp. 1636-1642, 1995.
- [4] Bylander T. (1994). The Computational Complexity of Propositional STRIPS Planning. In *Artificial Intelligence 69(1-2)*, p. 165-204, 1994.

- [5] Coles A., Fox M., Long D., Smith A. (2008). Planning with Problems Requiring Temporal Coordination. In *Proc. of AAAI 2008*, p. 892-897, 2008.
- [6] Cushing W., Kambhampati S., Mausam, Weld D.S. (2007). When is Temporal Planning Really Temporal? In *Proc. of 20th International Joint Conference on Artificial Intelligence*, p. 1852-1859, 2007.
- [7] Dean T., Firby J., Miller D., (1988). Hierarchical Planning involving deadlines, travel time and resources. In *Computational Intelligence 6(1)*, p. 381-398, 1988.
- [8] Fox M., Long D. (2003). PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. In *Journal of Artificial Intelligence Research 20*, p. 61-124, 2003.
- [9] Ghallab M. & Alaoui A.M. (1989). Managing Efficiently Temporal Relations Through Indexed Spanning Trees. In *Proc. of 11th International Joint Conference on Artificial Intelligence*, p. 1297-1303, 1989.
- [10] Halsey K., Long D., Fox M., (2004). CRIKEY - A Planner Looking at the Integration of Scheduling and Planning. In *Proc. of the "Integration Scheduling Into Planning" Workshop, ICAPS'03*, pp. 46-52, 2004.
- [11] Hu Y. (2007). Temporally-Expressive Planning as Constraint Satisfaction Problems. In *Proc. of 20th International Joint Conference on Artificial Intelligence*, p. 192-199, 2007.
- [12] Huang R., Chen Y., Zhang W. (2009). An Optimal Temporally Expressive Planner: Initial Results and Application to P2P Network Optimization. In *Proc. of 19th International Conference on Automated Planning and Scheduling*, 2009.
- [13] Laborie P. & Ghallab M. (1995) Planning with Sharable Resource Constraints. In *Proc. of 14th International Joint Conference on Artificial Intelligence*, p. 1643-1651, 1995.
- [14] Long D. & Fox M. (2003). Exploiting a graphplan framework in temporal planning. In *Proc. of 13th International Conference on Automated Planning and Scheduling*, p. 52-61, 2003.
- [15] Maris F. & Régnier P. (2008a). TLP-GP: Solving Temporally-Expressive Planning Problems. In *Proc. of 15th International Symposium on Temporal Representation and Reasoning*, p. 137-144, 2008.
- [16] Maris F. & Régnier P. (2008b). TLP-GP: New Results on Temporally-Expressive Planning Benchmarks. In *Proc. of 20th IEEE International Conference on Tools with Artificial Intelligence*, vol. 1, p. 507-514, 2008.
- [17] Maris F. & Régnier P. (2010). Planification temporellement expressive. Dans *Revue d'Intelligence Artificielle*, n° spécial "Planification, Décision et Apprentissage", à paraître, 2010.
- [18] McDermott D.V. (1998). PDDL, The Planning Domain Definition Language. In *Technical Report*. <http://cs-www.cs.yale.edu/homes/dvm/>
- [19] Muscettola N., (1994). HSTS: integrating planning and scheduling. In M. Zweben & M. Fox Eds. *Intelligent Scheduling*, p. 169-212, 1994.
- [20] Reichgelt H. & Shadbolt N. (1990). A Specification Tool for Planning Systems. In *Proc. of 9th European Conference on Artificial Intelligence*, p. 541-546, 1990.
- [21] Rintanen J. (2007). Complexity of Concurrent Temporal Planning. In *Proc. of 17th International Conference on Automated Planning and Scheduling*, p. 280-287, 2007.
- [22] Rutten E. & Hertzberg J. (1993). Temporal Planner = Nonlinear Planner + Time Map Manager. In *Artificial Intelligence Communications 6(1)*, p. 18-26, 1993.
- [23] Schwartz P. & Pollack M.E. (2004). Planning with Disjunctive Temporal Constraints. In *Proc. of ICAPS'04 Workshop on Integrating Planning into Scheduling*, 2004.
- [24] Tarjan R. (1972). Depth-first search and linear graph algorithms. In *SIAM Journal on Computing 1(2)*, p. 146-160, 1972.
- [25] Tsang E. (1987). TLP—a temporal planner. In *Advances in artificial intelligence*, p. 63-78, 1987.
- [26] Shin J. & Davis E. (2004). Continuous Time in a SAT-Based Planner. In *Proc. of 19th National Conference on Artificial Intelligence (AAAI'04)*, p. 531-536, 2004.
- [27] Wolfman S. & Weld D., (1999). The LPSAT Engine and its Application to Resource Planning. In *Proc. of 16th International Joint Conference on Artificial Intelligence*, p. 310-317, 1999.
- [28] Younes H.L.S. & Simmons R.G. (2003). VHPOP: Versatile Heuristic Partial Order Planner. In *Journal of Artificial Intelligence Research*, 20, p. 405-430, 2003.

Représentation des connaissances dans Flatland

Philippe Balbiani

Olivier Gasquet

François Schwarzentruher

IRIT, Université Paul Sabatier,
118 Route de Narbonne, F-31062 TOULOUSE CEDEX 9, FRANCE

balbiani@irit.fr
gasquet@irit.fr
schwarze@irit.fr

Résumé

Dans ce papier, nous étudions une sémantique concrète de la logique épistémique : des agents (personnes, robots, caméras etc.) et des objets (lampes, etc.) sont placés dans le plan et les propositions décrivent les états des objets (lampes allumées ou éteintes etc.) Un placement des objets et des agents donnent une situation épistémique. On utilisera le langage de la logique épistémique classique pour exprimer des propriétés comme “L’agent a sait que l’agent b sait que la lampe ℓ est allumée”. Les applications de cette logique pourraient être nombreuses : jeux vidéos, robotique, etc.

Mots clefs

Logique épistémique. Raisonnement spatial.

Abstract

We study a concrete semantics for epistemic logic : agents (humans, robots, etc.) and objects (lamps, etc.) are located in the plane and propositions describe states of objects (lamp on or off etc.) We use epistemic modal logic to express properties like “agent a knows that b knows that the lamp ℓ is on”. Applications can be numerous : video games, robotics etc.

Keywords

Epistemic modal logic. Spatial reasoning.

1 Introduction

Débutons ce papier en rappelant l’exemple des enfants sales [2] : des enfants au front sale se voient tous. Ils savent qu’au moins l’un d’eux est sale. En parlant tour à tour, chacun doit deviner s’il est sale ou non. L’étude de ce problème [2] donne vraiment l’impression qu’on résout le problème en le formalisant. En effet, on doit créer de toute pièce le modèle de Kripke qui représente la situation épistémique. Dans notre approche, la situation épistémique est directement représentée par un dessin (Figure 1).

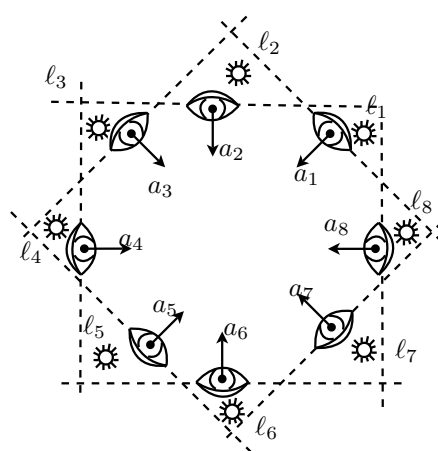


FIGURE 1 – Enfants sales dans Flatland

Sur certains points, notre approche peut être comparée à l’approche pédagogique de [1] où ils placent des cubes, pyramides et autre objets puis expriment des propriétés sur ces derniers avec la logique du premier ordre. Notre approche est similaire : on place des lampes et agents dans Flatland et on exprime des propriétés avec la logique épistémique.

Les applications de cette démarche pourraient être la robotique et les jeux vidéos, surtout les jeux de stratégies ou les jeux de rôles. En effet, les personnages (ennemis, etc.) de tels jeux évoluent dans des mondes de type Flatland et les comportements de ces personnages dépendent fortement de leur connaissance. Par exemple, le gardien du chateau devient fou s’il sait que la porte du chateau est ouverte. D’ailleurs, dans ce domaine, [4] donne une formalisation préliminaire du jeu vidéo Thief.

Vous noterez une différence intéressante entre la sémantique traditionnelle de la logique épistémique et celle qu’on donne ici : elle concerne la connaissance commune. Ici, si l’agent a sait que l’agent b sait que la lampe ℓ est allumée et que l’agent b sait aussi que l’agent a sait que la lampe ℓ est allumée, alors il y a connaissance commune entre les agents a et b que la lampe ℓ est allumée. On discutera de ce

fait dans la section 6.

2 Syntaxe

Dans cette section, nous introduisons un langage très similaire à celui de la logique épistémique classique $S5_n$ [3]. Soit $\text{LMP} = \{\ell, m, \dots\}$ un ensemble dénombrable de lampes. Soit $\text{AGT} = \{a, b, c, \dots\}$ un ensemble dénombrable d'agents. On notera $\text{OBJ} = \text{LMP} \cup \text{AGT}$. Les lampes et les agents sont des objets.

Définition 1 (langage)

Le langage \mathcal{L} est défini par la BNF suivante :

$$\varphi ::= \text{on}_\ell \mid \perp \mid \neg\varphi \mid (\varphi \vee \varphi) \mid K_a\psi$$

où $\ell \in \text{LMP}$ and $a \in \text{AGT}$.

Comme toujours on définit $(\varphi \wedge \psi) =_{\text{def}} \neg(\neg\varphi \vee \neg\psi)$ et $\hat{K}_a\psi =_{\text{def}} \neg K_a\neg\psi$. On omettra les parenthèses dans les formules quand il n'y a pas ambiguïté. Soit $\text{Imp}(\varphi)$ l'ensemble des lampes dont parle la formule φ et $\text{agt}(\varphi)$ l'ensemble des agents dont parle la formule φ . Soit $\text{obj}(\varphi)$ l'ensemble des objets dont parle la formule φ . Bien sûr $\text{obj}(\varphi) = \text{agt}(\varphi) \cup \text{Imp}(\varphi)$. La formule $K_a\text{on}_\ell$ se lit "l'agent a sait que la lampe ℓ est allumée". La formule $K_a\neg\text{on}_\ell$ se lit "l'agent a sait que la lampe ℓ est éteinte".

3 Notations

Dans cette section, on rappelle des notions basiques de géométrie. On note \mathbb{N} l'ensemble des entiers naturels et \mathbb{R} l'ensemble des réels. L'ensemble \mathbb{R}^2 désigne le plan réel. Si $a \in \mathbb{R}^2$, on note $a = (a_x, a_y)$ où $a_x, a_y \in \mathbb{R}$: a_x est l'abscisse de a et a_y est l'ordonnée de a . Si $a, b \in \mathbb{R}^2$, on définit $\vec{ab} \in \mathbb{R}^2$ par $\vec{ab} = (b_x - a_x, b_y - a_y)$.

Le produit scalaire de a et b est défini par $a \cdot b = a_x \times b_x + a_y \times b_y \in \mathbb{R}$. Si $x \in \mathbb{R}^2$, on définit la norme euclidienne de x par $\|x\| = \sqrt{x \cdot x}$.

L'ensemble $U = \{x \in \mathbb{R}^2 \mid \|x\| = 1\}$ désigne le cercle unité.

4 Sémantique

La sémantique n'est pas définie à partir d'une classe de modèle mais par une situation dans Flatland. Un monde est une situation où chaque agent possède une localisation (position et direction de vue) et chaque lampe possède une position et un état. Formellement :

Définition 2 (monde)

Un monde w est un triplet $\langle \text{pos}, \vec{\text{dir}}, \text{val} \rangle$ où :

- $\text{pos} : \text{OBJ} \rightarrow \mathbb{R}^2$;
- $\vec{\text{dir}} : \text{AGT} \rightarrow U$;
- $\text{val} : \text{LMP} \rightarrow \{\text{off}, \text{on}\}$.

L'ensemble des mondes est noté W .

Dans un monde $w = \langle \text{pos}, \vec{\text{dir}}, \text{val} \rangle$, $\text{pos}(\beta)$ désigne la position de l'objet β (agent ou lampe) dans le plan. Le vecteur $\vec{\text{dir}}(a)$ est un vecteur de norme 1 qui désigne la direction

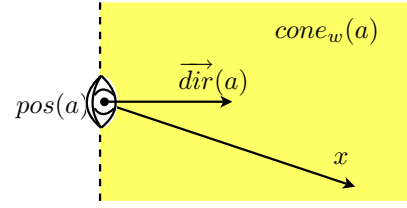


FIGURE 2 – $\text{cone}_w(a)$ = le demi-plan fermé vu par l'agent a

dans laquelle regarde l'agent a . L'agent voit alors tout un demi-plan fermé. Si $\text{val}(\ell) = \text{on}$ alors la lampe ℓ est allumée. Si $\text{val}(\ell) = \text{off}$ alors la lampe ℓ est éteinte.

Exemple 1 La figure 1 représente un monde dans le sens de la définition 2 :

- $\text{AGT} = \{a_1, \dots, a_8\}$; $\text{LMP} = \{\ell_1, \dots, \ell_8\}$;
- $\text{pos}(a_k) = (\cos(\frac{k\pi}{4}), \sin(\frac{k\pi}{4}))$; (positions des agents)
- $\text{pos}(\ell_k) = (1.1 \times \cos(\frac{k\pi}{4}), 1.1 \times \sin(\frac{k\pi}{4}))$; (positions des lampes ℓ_k)
- $\vec{\text{dir}}(a_k) = (-\cos(\frac{k\pi}{4}), -\sin(\frac{k\pi}{4}))$; (directions des agents)
- $\text{val}(\ell_k) = \text{on}$.

Désormais, on définit l'ensemble des points que l'agent a voit.

Définition 3 (cône)

Considérons un monde $w = \langle \text{pos}, \vec{\text{dir}}, \text{val} \rangle$.

Pour tout $a \in \text{AGT}$, on note $\text{cone}_w(a)$ l'ensemble $\{x \in \mathbb{R}^2 \mid \vec{\text{dir}}(a) \cdot \text{pos}(a)x \geq 0\}$.

Comme le montre la figure 2, $\text{cone}_w(a)$ est le demi-plan fermé des points x tels que $\vec{\text{dir}}(a) \cdot \text{pos}(a)x \geq 0$. On pourrait changer cette définition :

- un agent pourrait voir un demi-plan ouvert ;
- un agent pourrait voir un cône d'un angle α : $\text{cone}_w(a) = \{x \in \mathbb{R}^2 \mid \vec{\text{dir}}(a) \cdot \text{pos}(a)x \geq \cos(\alpha) \|\text{pos}(a)x\|\}$;
- un agent pourrait être myope et ne pas voir les choses au delà de la distance r : $\text{cone}_w(a) = \{x \in \mathbb{R}^2 \mid \vec{\text{dir}}(a) \cdot \text{pos}(a)x \geq 0 \text{ and } \|\text{pos}(a)x\| \leq r\}$.

Ici, pour faire simple, on adopte la définition 3.

À partir de maintenant, définissons les relations épistémiques entre les mondes. Pour tout $w, u \in W$, $wR_a u$ signifie que l'agent a ne distingue pas le monde w du monde u , i.e. l'agent a voit les mêmes objets dans le monde w et dans le monde u . Les mondes w et u ne diffèrent que par les positions, directions et états des objets que l'agent a ne voit pas. Formellement,

Définition 4 (relation épistémique)

Soit $a \in \text{AGT}$. On définit la relation R_a entre deux mondes $w = \langle \text{pos}, \vec{\text{dir}}, \text{val} \rangle$ et $u = \langle \text{pos}', \vec{\text{dir}}', \text{val}' \rangle$: $wR_a u$ ssi :

- pour tout $\beta \in \text{OBJ}$,
- $$\text{pos}(\beta) \in \text{cone}_w(a) \text{ ssi } \text{pos}'(\beta) \in \text{cone}_u(a) ;$$

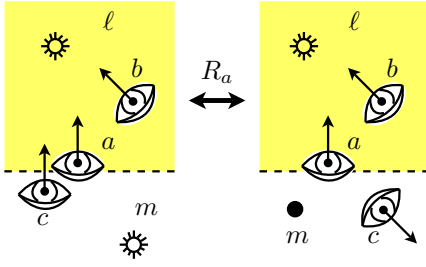


FIGURE 3 – Deux mondes reliés par R_a

- pour tout $b \in \text{AGT}$, si $\text{pos}(b) \in \text{cone}_w(a)$ alors
 $\text{pos}(b) = \text{pos}'(b)$ et $\vec{\text{dir}}(b) = \vec{\text{dir}}'(b)$;
- et pour tout $\ell \in \text{LMIP}$, si $\text{pos}(\ell) \in \text{cone}_w(a)$ alors
 $\text{pos}(\ell) = \text{pos}'(\ell)$ et $\text{val}(\ell) = \text{val}'(\ell)$.

La figure 3 montre deux mondes reliés par R_a : les objets a , b , ℓ vu par l'agent a ont les mêmes positions directions/états dans les deux mondes. Par contre, l'agent c et la lampe m ont changé de directions/états et positions car ils sont invisibles pour l'agent a .

Il est trivial que la relation R_a est une relation d'équivalence sur l'ensemble W .

Définition 5 (Conditions de vérité)

Soit $w = \langle \text{pos}, \vec{\text{dir}}, \text{val} \rangle \in W$ et une formule φ . On définit $w \models \varphi$ par induction :

- $w \not\models \perp$;
- $w \models \text{on}_\ell$ ssi $\text{val}(\ell) = \text{on}$;
- $w \models \varphi \vee \psi$ ssi $w \models \varphi$ ou $w \models \psi$;
- $w \models \neg\varphi$ ssi $w \not\models \varphi$;
- $w \models K_a\psi$ ssi pour tout $u \in W$, $wR_a u$ implique $u \models \psi$.

On dira qu'une formule φ est *satisfiable* ssi il existe un monde $w \in W$ tel que $w \models \varphi$. Une formule φ est *valide* ssi pour tout monde $w \in W$, $w \models \varphi$. Comme R_a est une relation d'équivalence sur W , les axiomes de la logique épistémique sont valides :

- $K_a\varphi \rightarrow \varphi$;
- $K_a\varphi \rightarrow K_aK_a\varphi$;
- $\neg K_a\varphi \rightarrow K_a\neg K_a\varphi$.

Les formules suivantes sont également valides :

- $K_a(\text{on}_\ell \vee \text{on}_m) \rightarrow K_a\text{on}_\ell \vee K_a\text{on}_m$;
- $K_aK_b\text{on}_\ell \wedge K_bK_a\text{on}_\ell \rightarrow K_aK_bK_a\text{on}_\ell$.

5 Deux problèmes de décision

Dans cette section, nous allons nous intéresser aux deux problèmes suivants.

Définition 6 (model-checking dans Flatland)

Le *model-checking dans Flatland* est le problème suivant :

- Entrée : une formule $\varphi \in \mathcal{L}$, et la *description du monde* w ;
- Sortie : Oui si on a $w \models \varphi$. Non, sinon.

Dans la définition 6, seul les objets qui apparaissent dans la formule φ sont pris en compte dans la *description d'un monde* w . Les positions sont supposées avoir des *coordonnées rationnelles* car on a besoin de représenter les positions en machine. De la même façon, une direction¹ $\vec{\text{dir}}(a)$ est représentée par son abscisse $\vec{\text{dir}}(a)_x$, supposée rationnelle et le signe de l'ordonnée $\vec{\text{dir}}(a)_y$. La valeur de $\vec{\text{dir}}(a)_x$ et le signe de $\vec{\text{dir}}(a)_y$ déterminent entièrement $\vec{\text{dir}}(a)$ car $\vec{\text{dir}}(a)_y^2 = 1 - \vec{\text{dir}}(a)_x^2$.

Définition 7 (problème de satisfiabilité dans Flatland)

Le *problème de satisfiabilité dans Flatland* est le problème suivant :

- Entrée : une formule $\varphi \in \mathcal{L}$;
- Sortie : oui s'il existe un monde w tel que $w \models \varphi$. Non, sinon.

Theorème 1 *Le model-checking dans Flatland et le problème de satisfiabilité dans Flatland sont décidables.*

PROOF.

On peut réduire ces deux problèmes au problème de la satisfiabilité d'une certaine formule dans la théorie des réels qui est décidable[5]. ■

6 Connaissance commune dans flatland

Dans cette section, on ajoute un opérateur de *connaissance commune* au langage, i.e., si φ est une formule alors $CK_J\varphi$ est aussi une formule pour tout sous-ensemble fini J d'agents. On a $w \models CK_J\varphi$ ssi pour toute séquence finie d'agents (a_1, \dots, a_n) de J , on a $w \models K_{a_1} \dots K_{a_n}\varphi$. Voici une caractérisation de $w \models K_{a_1} \dots K_{a_n}\text{on}_\ell$:

Proposition 1 *Soit $n \in \mathbb{N}$, $a_1, \dots, a_n \in \text{AGT}$ et $\ell \in \text{LMIP}$. Soit $w \in W$. On a équivalence entre :*

1. $w \models K_{a_1} \dots K_{a_n}\text{on}_\ell$;
2. – pour tout $j \in \{1, \dots, n\}$, $\text{pos}(\ell) \in \text{cone}_w(a_j)$;
– pour tout $j, k \in \{1, \dots, n\}$, $j < k$ implique $\text{pos}(a_k) \in \text{cone}_w(a_j)$;
– et $\text{val}(\ell) = \text{on}$.

Du coup, on a une caractérisation géométrique de $w \models CK_JK_{b_1} \dots K_{b_k}\text{on}_\ell$:

Proposition 2 *Soit J un sous-ensemble fini de AGT , $\ell \in \text{LMIP}$ et $w \in W$. On a $w \models CK_JK_{b_1} \dots K_{b_k}\text{on}_\ell$ dans *iff* (*)*

- $\text{val}(\ell) = \text{on}$;
- pour tout $a \in J$, $\text{pos}(\ell) \in \text{cone}_w(a)$;
- pour tout $i \in \{1, \dots, k\}$, $\text{pos}(\ell) \in \text{cone}_w(b_i)$;
- pour tout $a \in J$, $i \in \{1, \dots, k\}$, $\text{pos}(b_i) \in \text{cone}_w(a)$;
- pour tout $a, b \in J$, $\text{pos}(b) \in \text{cone}_w(a)$;
- et pour tout $i, j \in \{1, \dots, k\}$, $i < j$ implique $\text{pos}(b_j) \in \text{cone}_w(b_i)$.

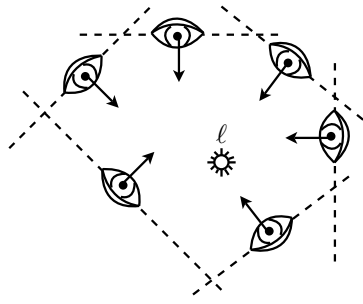


FIGURE 4 – Un monde w tel que $w \models CK_J on_\ell$

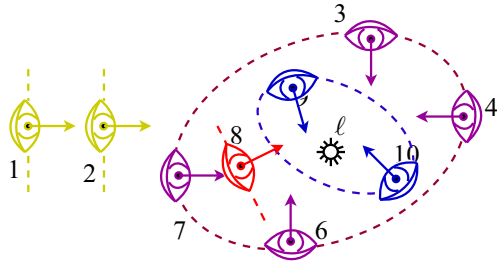


FIGURE 5 – $K_1 K_2 CK_{\{3,4,6,7\}} K_8 CK_{\{9,10\}} on_\ell$

D'où $CK_J K_{b_1} \dots K_{b_k} on_\ell$ est équivalent à des formules sans opérateur de connaissance commune CK_J :

Corollaire 1 Soit $J = \{a_1, \dots, a_n\}$ un sous-ensemble fini de AGT , $b_1, \dots, b_k \in \text{AGT}$, $\ell \in \text{LMP}$ et $w \in W$. On a équivalence entre :

- $w \models CK_J K_{b_1} \dots K_{b_k} on_\ell$;
- $w \models K_{a_1} \dots K_{a_n} K_{a_1} \dots K_{a_n} K_{b_1} \dots K_{b_k} on_\ell$;
- $w \models \bigwedge_{a,b \in J} K_a K_b K_{b_1} \dots K_{b_k} on_\ell$.

La figure 4 représente un monde où il y a connaissance commune de on_ℓ . La figure 5 représente un monde où $K_1 K_2 CK_{\{3,4,6,7\}} K_8 CK_{\{9,10\}} on_\ell$ est vraie. En appliquant la proposition 1, la formule $K_1 K_2 CK_{\{3,4,6,7\}} K_8 CK_{\{9,10\}} on_\ell$ est équivalente à des contraintes géométriques sur les objets comme le montre la figure 5.

Les décidabilités du model-checking et problème de la satisfiabilité des formules avec l'opérateur de connaissance commune sont ouvertes.

7 Perspectives

Les perspectives sont nombreuses :

- Du travail de modélisation : avoir des lampes plus expressives (une lampe = une formule et pas seulement une proposition) ;
- Du travail théorique : trouver la complexité du model-checking et du problème de la satisfiabilité des formules dans Flatland ainsi qu'une axiomatisation ;

1. Nous n'allons pas représenter une direction par un angle... car la théorie des réels + la fonction cos est... indécidable ! En effet, on peut y définir les entiers et donc toute l'arithmétique de Peano.

- Du travail pratique : implémenter un solveur pour les deux problèmes énoncés plus haut.

Remerciements. Merci à Andreas Herzig et Emiliano Lorini pour leurs précieux conseils.

Références

- [1] Jon Barwise and John Etchemendy. *Tarski's World : Version 4.0 for Macintosh (Center for the Study of Language and Information - Lecture Notes)*. Center for the Study of Language and Information/SRI, 1993.
- [2] Jelle Gerbrandy. *Bisimulations on Planet Kripke*. PhD thesis, University of Amsterdam, 1999.
- [3] Joseph Y. Halpern and Yoram Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54(3) :319–379, 1992.
- [4] Ethan Kennerly, Andreas Witzel, and Jonathan A. Zvesper. Thief belief (extended abstract). Presented at Logic and the Simulation of Interaction and Reasoning 2 (LSIR2) Workshop at IJCAI-09, July 2009.
- [5] Alfred Tarski. A decision method for elementary algebra and geometry. 1951.

Représentation temporelle qualitative de recettes de cuisine

F. Le Ber^{1,2}

J. Lieber²

A. Napoli²

¹ LHYGES UMR 7517

² LORIA UMR 7503

¹ ENGEES, 1 quai Koch, 67000 Strasbourg

florence.leber@engees.unistra.fr

² Campus scientifique, 54500 Vandœuvre-lès-Nancy

{Jean.Lieber, Amedeo.Napoli}@loria.fr

Résumé

Le projet TAAABLE a pour objet de construire un système de raisonnement à partir de cas pour la recherche et l'adaptation de recettes de cuisine. Dans ce cadre nous nous intéressons aux aspects temporels des recettes et à la manière de les représenter afin d'en envisager l'adaptation. Notre ambition est plus généralement d'étudier l'apport des modèles temporels qualitatifs au raisonnement à partir de cas.

Mots Clef

Modèles qualitatifs du temps, cuisine, raisonnement à partir de cas, TAAABLE.

Abstract

The aim of the TAAABLE project is to develop a case-based reasoning system for the retrieval and the adaptation of cooking recipes. In this paper, we focus on the temporal features of cooking recipes and their representation in order to help the adaptation process. Our more general objective is to study the introduction of qualitative temporal models within case-based reasoning.

Keywords

Qualitative temporal models, cooking, case-based reasoning, TAAABLE.

1 Introduction

Cet article est une réflexion préliminaire sur l'utilisation de modèles qualitatifs du temps dans le raisonnement à partir de cas. Nous prenons pour exemple d'application le projet TAAABLE [3], sur la représentation, la remémoration et l'adaptation de recettes de cuisine. L'objectif de cette application est de générer des recettes opérationnelles, à partir d'une liste d'ingrédients et de contraintes, en s'appuyant sur une base de recettes données. L'article est focalisé principalement sur la question de la représentation des cas. Il est organisé comme suit. Une première partie présente les principes du raisonnement à partir de cas et

le cadre du projet TAAABLE. Une deuxième partie détaille un exemple qui met en évidence quelques problématiques intéressantes pour les modèles qualitatifs du temps. Enfin nous concluons en évoquant quelques pistes de recherche.

2 Raisonner à partir de recettes de cuisine

2.1 Le raisonnement à partir de cas

Le raisonnement à partir de cas (noté RàPC) est un mode de résolution de problèmes s'appuyant sur la réutilisation de solutions de problèmes déjà résolus [12]. Pour une application particulière du RàPC, on considère les notions de problème et de solution : résoudre un problème c'est lui associer une solution. Un cas est la représentation d'un épisode de résolution de problème : il encode un problème et une solution de ce problème (à laquelle sont généralement associées des informations sur les liens entre ce problème et cette solution). L'ensemble des cas dont un système de RàPC dispose s'appelle base de cas et un cas de cette base est appelé cas source (dont la partie problème est appelé problème source). Raisonner à partir de cas, c'est résoudre un problème, appelé problème cible en s'appuyant sur une base de cas et, en général, sur des connaissances particulières au domaine d'application.

Une session de RàPC est généralement composée de deux étapes principales : la *remémoration* et l'*adaptation*. La remémoration consiste à sélectionner un cas source jugé « similaire » au problème cible. L'adaptation a alors pour objectif de s'appuyer sur le cas source remémoré pour résoudre le problème cible, souvent en modifiant la solution associée au cas source sur la base d'une représentation de la différence entre le problème source et le problème cible. De façon générale, l'adaptation consiste à résoudre un problème de raisonnement par analogie : connaissant les problèmes source et cible, les liens entre ces deux problèmes (ressemblances, différences) et les liens entre le problème source et sa solution (résolution de problème), on cherche à établir une solution du problème cible. Parfois, une étape

de *mémorisation* suit la résolution de problème et consiste à mémoriser, si c'est jugé opportun, le cas formé par le problème cible et sa solution.

2.2 Le projet TAAABLE

Le *Computer Cooking Contest*¹ (CCC) est un concours organisé annuellement depuis 2008. Un candidat au CCC est un système informatique destiné à résoudre des problèmes de cuisine à partir d'un livre de recettes, chaque recette étant donnée par un fichier XML (voir section suivante pour un exemple). Le projet TAAABLE s'est mis en place dans le but initial de concourir au CCC ; il a été classé deuxième au CCC-2008 et au CCC-2009. Son deuxième objectif est de fédérer des recherches autour de la construction de systèmes à base de connaissances (représentation des connaissances, RàPC, acquisition et extraction de connaissances, fouille de textes, wikis sémantiques, etc.).

En tant que système de RàPC, le système TAAABLE considère les recettes comme des cas, les requêtes à poser au système (tels que « J'aimerais une recette de dessert avec de la pomme et sans cannelle ») comme des problèmes. Dans les versions actuelles de ce système [3, 4] — et des autres systèmes ayant participé aux CCC-2008 et 2009 — les inférences ne portaient que sur les ingrédients. Les inférences sur la partie « préparation » des recettes constituent une étude en cours, dans laquelle la représentation temporelle doit jouer un grand rôle.

3 Un exemple

Prenons un exemple tiré de la base de recettes du premier CCC (toutes les recettes de cette base ont la même forme). C'est une recette de gâteau (ou cake salé) aux courgettes (voir ci-dessous). La recette est divisée en deux parties, la première (introduite par la balise <IN>) listant les ingrédients, la deuxième (introduite par la balise <PR>) listant des actions ou opérations. Dans cette deuxième partie, le temps est exprimé de manière qualitative ou quantitative. On remarque tout d'abord que la structure même du texte indique un ordre dans les actions. De plus, des marqueurs temporels sont utilisés pour exprimer par exemple des durées de cuisson : « cuire *pendant* 5 minutes ou *jusqu'à* ce que les courgettes soient tendres ».

```
<RECIPE>
<TI>Zucchini, Chile Corn Bake</TI>
<IN>1 tb Vegetable oil</IN>
<IN>1 lb Zucchini; grated</IN>
<IN>1/2 c Chopped onion</IN>
<IN>3 Eggs</IN>
<IN>3 c Cooked rice</IN>
<IN>7 oz Whole kernel corn (canned) drained</IN>
<IN>8 oz Chopped green chilies</IN>
<IN>2 c Cheddar cheese, grated</IN>
<IN>4 oz Crumbled queso fresco OR
very mild feta</IN>
<IN>1 ts Salt</IN>
<IN>Vegetable cooking spray</IN>
```

¹<http://www.wi2.uni-trier.de/coc09>

```
<PR>Heat oil in large skillet over medium heat
until hot. Add zucchini and onion; cook
uncovered, stirring constantly, for 5
minutes or until zucchini is soft.
Remove from heat; set aside.
[Then] Beat eggs in large bowl. Stir
in rice, corn chiles, cheese, zucchini
mixture, and salt. Mix well together.
[Then] Pour into 13 x 9 x 2-inch
baking pan coated with cooking spray.
Bake at 375 degrees 45 to 50 minutes
or until knife inserted in center
comes out clean.
</PR>
</RECIPE>
```

La séquence d'actions (deuxième partie de la recette) peut être représentée par le graphe temporel de la figure 1 (partie gauche). Le graphe est composé de trois actions principales qui se succèdent linéairement, ce qui peut être représenté par la relation $\{b, m\}$ ² de l'algèbre des intervalles [1, 2]. Chaque action peut être décomposée en un sous-graphe, comme montré pour l'action « cuire les oignons et les courgettes ». La transcription du texte sous la forme d'un tel graphe soulève un certain nombre de difficultés relevant du traitement du langage naturel, difficultés dans lesquelles nous n'entrons pas ici. Nous insisterons davantage sur la représentation des éléments temporels, en nous limitant pour l'instant aux informations qualitatives.

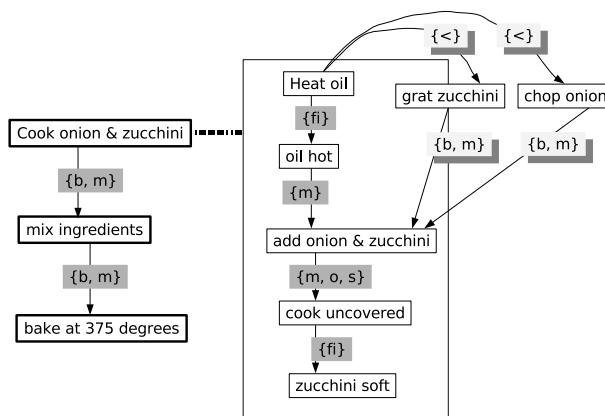


FIG. 1 – Préparation d'un gâteau aux courgettes : graphe temporel décrivant la recette avec ajout de relations par utilisation des connaissances du domaine et d'inférences temporelles.

Si on regarde plus attentivement la partie « ingrédients » de la recette, on remarque que certains des ingrédients nécessitent une préparation. Ainsi, les courgettes et les oignons sont émincés. Cette préparation doit avoir lieu avant les actions décrites dans la deuxième partie de la recette. Dans le graphe représenté figure 1 (partie droite), nous ajoutons ces préparations et indiquons qu'elles doivent être réalisées avant l'action « ajouter l'oignon et les courgettes » (relation $\{b, m\}$). En revanche seule

² b dénote la relation *before* entre deux intervalles de temps, m la relation *meets*.

une connaissance du domaine (il est plus long d'émincer des courgettes que de chauffer de l'huile) nous permet d'établir une relation temporelle entre ces préparations et l'action « chauffer l'huile ». Nous utilisons ici le modèle *INDU* [11] qui permet de manipuler à la fois les relations de durée et les relations topologiques entre intervalles. Ainsi, à partir de la relation sur les durées³, qui est un axiome de la base de connaissances :

$$(1) I_{\text{heat-oil}} \{<\} I_{\text{grat-zucchini}}$$

et des relations lues sur le graphe :

$$(2) I_{\text{heat-oil}} \{f_i\} I_{\text{oil-hot}}$$

$$(3) I_{\text{oil-hot}} \{m\} I_{\text{add-zucchini}}$$

$$(4) I_{\text{grat-zucchini}} \{b, m\} I_{\text{add-zucchini}}$$

On peut inférer que l'éminçage des courgettes doit débiter avant la mise en chauffe de l'huile – la relation entre les deux intervalles de temps correspondants est une des relations *before*, *meets*, *overlaps*, *finished-by*, combinées avec la relation sur les durées conformément au modèle *INDU* :

$$(5) I_{\text{grat-zucchini}} \{b^>, m^>, o^>, f_i^>\} I_{\text{heat-oil}}$$

Le but ici n'est pas forcément de détecter toutes les relations possibles entre les actions (en saturant le graphe) mais plutôt de pouvoir proposer un scénario à l'utilisateur, lui indiquant les opérations à effectuer dans un ordre correct. Une évaluation des durées de ces opérations peut également être envisagée, en utilisant des intervalles de durées fixées et ordonnées, ce que permet le formalisme *INDU*. Par exemple on pourrait écrire que :

$$I_{\text{cook-uncovered}} \{eq^=\} I_{5mn}$$

pour signifier que le temps de cuisson est de cinq minutes. Les informations quantitatives exprimées dans la recette peuvent alors être combinées avec les relations temporelles qualitatives établies dans le graphe.

4 Discussion et perspectives

Notons tout d'abord qu'il existe différents travaux intégrant des aspects temporels dans un modèle de RàPC mais que fort peu s'intéressent à des modèles qualitatifs du temps. Seuls les travaux de [7] et de [5], à notre connaissance, utilisent une représentation du temps à base d'intervalles et de relations de Allen. Parmi ces travaux, certains s'inscrivent dans le domaine de la planification à partir de cas, qui est un champ de recherche actif du RàPC, et dont l'un des premiers systèmes, le système CHEF, s'applique précisément à la planification de recettes de cuisine [6]. On pourra aussi s'intéresser aux travaux touchant à la reconnaissance de

³ I_{action} représente l'intervalle de temps pendant lequel se déroule l'action, $<$ dénote la relation « est plus court que ».

plans, comme par exemple [8], qui fait appel – ici aussi avec une application dans le domaine culinaire – à la fois à la modélisation d'actions et au raisonnement temporel. Un des objectifs de notre travail est ainsi de réfléchir à l'intégration des mécanismes de raisonnement temporel qualitatif dans le modèle du RàPC, comme nous l'avons fait pour le raisonnement spatial qualitatif dans le cadre d'une application agronomique [10].

En ce qui concerne le projet TAAABLE, la mise en œuvre des modèles qualitatifs du temps pose différents problèmes. Actuellement, les cas sont représentés comme une liste d'ingrédients, et la remémoration est indépendante de toute représentation temporelle. Par exemple si on veut faire un cake salé et qu'on dispose d'aubergines, le système proposera la recette de cake aux courgettes parce que aubergines et courgettes sont proches dans l'ontologie du domaine. Pour prendre en compte les aspects temporels, il faut d'abord transcrire automatiquement les recettes en graphes temporels, puis établir les connaissances du domaine (sur les temps de cuisson, l'ordre des opérations, etc.). En supposant que la remémoration s'appuie toujours et uniquement sur une liste d'ingrédients, son résultat comprendra alors un graphe temporel associé à la recette remémorée. La procédure d'adaptation devra s'appuyer sur un raisonnement temporel pour construire une recette opérationnelle à partir de ce graphe (par exemple les aubergines sont plus longues à cuire que les courgettes, il faut rajouter de l'huile en cours de cuisson, etc.). Cette procédure pourra s'effectuer ainsi : le système étiquette d'abord le graphe représentant la recette remémorée avec les nouveaux ingrédients (en remplaçant les courgettes par les aubergines) puis adapte la structure du graphe, c'est-à-dire modifie les étiquettes des relations et éventuellement ajoute ou supprime des nœuds et des arêtes. Ces modifications peuvent conduire à des incohérences qu'il s'agira ensuite de résoudre. Quelques exemples illustrant cette procédure sont décrits dans [9].

Ces différentes questions renvoient aux méthodes développées dans le domaine des réseaux de contraintes qualitatives mais aussi en planification et en révision de croyances, méthodes que nous nous proposons d'étudier et de tester dans le cadre d'un prochain *Computer Cooking Contest*.

Références

- [1] Allen (J. F.). – An interval-based representation of temporal knowledge. In : *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI'81)*, pp. 221–226.
- [2] Allen (J. F.). – Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, vol. 26, n11, 1983, pp. 832–843.
- [3] Badra (F.), Bendaoud (R.), Bentebitel (R.), Champin (P.-A.), Cojan (J.), Cordier (A.), Desprès (S.), Jean-Daubias (S.), Lieber (J.), Meilender (T.), Mille (A.), Nauer (E.), Napoli (A.) et Toussaint (Y.). – Taaable :

Text Mining, Ontology Engineering, and Hierarchical Classification for Textual Case-Based Cooking. *In : ECCBR Workshops, Workshop of the First Computer Cooking Contest*, pp. 219–228.

- [4] Badra (F.), Cojan (J.), Cordier (A.), Lieber (J.), Meilender (T.), Mille (A.), Molli (P.), Nauer (E.), Napoli (A.), Skaf Molli (H.) et Toussaint (Y.). – Knowledge acquisition and discovery for the textual case-based cooking system WIKITAAABLE. *In : 8th International Conference on Case-Based Reasoning - ICCBR 2009, Workshop Proceedings*, pp. 249–258. – Seattle, 2009.
- [5] Dørum Jære (M.), Aamodt (A.) et Skalle (P.). – Representing temporal knowledge for case-based prediction. *In : Advances in Case-Based Reasoning, Proceedings of ECCBR 2002*. pp. 174–188. – Springer-Verlag.
- [6] Hammond (K. J.). – Case-Based Planning : A Framework for Planning from Experience. *Cognitive Science*, vol. 14, n3, 1990, pp. 385–443.
- [7] Jaczynski (M.). – Modèle et plate-forme à objets pour l’indexation par situations comportementales : application à la navigation sur le Web. – Thèse de doctorat, Université de Nice-Sophia-Antipolis, décembre 1998.
- [8] Kautz (H. A.) et Allen (J. F.). – Generalized plan recognition. *In : AAAI-86 Proceedings*, pp. 32–37.
- [9] Le Ber (F.), Lieber (J.) et Napoli (A.). – Utilisation d’un algèbre temporelle pour la représentation et l’adaptation de recettes de cuisine. *In : Actes du 17ème Séminaire Raisonnement à partir de Cas, RàPC 2009, Paris*, pp. 141–149.
- [10] Le Ber (F.), Napoli (A.), Metzger (J.-L.) et Lardon (S.). – Modeling and comparing farm maps using graphs and case-based reasoning. *Journal of Universal Computer Science*, vol. 9, n9, septembre 2003, pp. 1073–1095.
- [11] Pujari (A. K.), Kumari (G. V.) et Sattar (A.). – INDU : an interval and duration network. *In : Australian Joint Conf. on Artificial Intelligence*, pp. 291–303.
- [12] Riesbeck (C. K.) et Schank (R. C.). – *Inside Case-Based Reasoning*. – Hillsdale, New Jersey, Lawrence Erlbaum Associates, Inc, 1989.

Analyse temps réel du comportement d'objets mobiles évoluant dans un espace ouvert

Laurent ETIENNE¹

Thomas DEVOGELE¹

Alain BOUJU²

¹ Institut de Recherche de l'Ecole Navale

² Laboratoire Informatique, Image et Interaction

Institut de Recherche de l'Ecole Navale
Lanvéoc-Poulmic, CC 600, 29240 BREST Cedex 9
laurent.etienne@ecole-navale.fr

Résumé

De plus en plus d'objets mobiles sont équipés de capteurs qui facilitent la surveillance de leur évolution et permettent la réception, le stockage et la visualisation en temps réel de leur position. Actuellement, la fouille de données dans des historiques de déplacements volumineux offre des perspectives intéressantes dans le domaine de l'étude du comportement de ces objets. À partir de résultats d'analyses statistiques sur des ensembles de trajectoires, des patrons spatio-temporels (trajectoires médianes, couloirs spatio-temporels, routes types) ont été définis. En s'appuyant sur ces patrons, il est possible de détecter en temps réel des situations inhabituelles et d'en qualifier le comportement. Dans cet article, cette méthode est appliquée aux déplacements de navires dans un espace maritime ouvert afin de détecter des comportements inhabituels (retard, hors route, etc.).

Mots Clef

trajectoires d'objets mobiles, similitude, trajectoire médiane, couloir spatio-temporel, comportement anormal.

Abstract

More and more mobile objects are now equipped with sensors allowing real time monitoring of their movements. Nowadays, the data produced by these sensors is received and stored in spatio-temporal databases in order to visually follow their movements. The main goal of this article is to perform a data mining on a huge quantity of stored positions in order to deduce the behaviour of these mobile objects. From results of statistical analyses on sets of trajectories of mobile objects following the same itinerary, we can compute spatio-temporal patterns (main routes and spatio-temporal channels). Using these patterns, unusual situations can be detected in real time and qualified. In this article, this technique is applied to ships' movements in an open maritime area in order to detect unusual behaviours.

Keywords

mobile objects, trajectories, similarity, median trajectory, spatio-temporal channel, unusual behaviour.

1 Introduction

La fouille de données (data mining) est employée classiquement pour extraire des connaissances à partir de grandes quantités de données par des méthodes automatiques ou semi-automatiques. Cette fouille de données, appliquée à des historiques volumineux de déplacements d'objets mobiles, offre des perspectives intéressantes dans le domaine de l'étude du comportement de ces objets. Les différentes analyses de similarité de trajectoires proposées dans la littérature se basent sur la densité, la distance ([14, 4, 7]) ou la forme ([10]) entre des trajectoires afin de définir des clusters de trajectoires et d'en déduire des trajectoires aberrantes ([1]). Lee et al. ([9, 3]) ont réalisé une bibliographie importante sur la détection de trajectoires inhabituelles par data mining. Il est à noter que les travaux mentionnés prennent en compte essentiellement l'aspect spatial. La partie temporelle de l'analyse est un sujet de recherche actif encore ouvert ([4, 13]). En ce qui concerne nos recherches, nous partons du postulat que des objets mobiles d'un même type suivant un même itinéraire ont des comportements similaires et suivent une route optimisée en termes de sécurité, de coût et de temps. Par conséquent, des comportements types peuvent être définis ([8]).

La section suivante de cet article présente et formalise les différents concepts employés. Dans la troisième section sont exposées les méthodes de création des patrons spatio-temporels. Enfin, la dernière section traite de l'analyse temps réel du comportement d'un objet mobile en se basant sur l'utilisation des patrons spatio-temporels préalablement définis. Une étude de cas portant sur les navires à passagers en rade de Brest illustre les différentes sections.

2 Modélisation des itinéraires et des trajectoires

2.1 Trajectoire d'un objet mobile

Les capteurs placés sur des objets mobiles génèrent une quantité importante de données de position. Celles-ci permettent pour chaque objet identifié de disposer à différents instants de leur localisation et retracer leur trajectoire. Afin d'analyser les déplacements des objets mobiles, ces suites de positions sont regroupées afin de composer la trajectoire T_o de l'objet mobile o . Cette trajectoire peut être formellement définie comme une suite de positions $P_{oj} = (x_j, y_j, t_j)$ ordonnées temporellement $T_o = (P_{o1}, \dots, P_{oj}, \dots, P_{on})$. Dans notre cas d'étude maritime, le système AIS équipant les navires est destiné à la sécurité du trafic maritime, il transmet la position du navire à intervalle variable en fonction de la vitesse et du déplacement du navire ([8]).

2.2 Graphe de zones

Afin d'analyser finement l'historique de ces données de position, la première étape du processus consiste à modéliser les informations implicites que sont les itinéraires et les trajectoires. Contrairement aux objets mobiles se déplaçant sur un réseau, les navires, comme les avions, ou les piétons, se déplacent dans un espace ouvert. Cependant, un réseau implicite existe. En effet, cet espace ouvert peut comporter des obstacles (physiques ou liés à la réglementation) ou des zones d'intérêt formées des destinations des objets mobiles et des lieux de passage obligatoires ou réglementés. Dans notre cas d'étude, les ports, les détroits, les dispositifs de séparation du trafic (DST) tels que le rail d'Ouessant, les zones d'attente forment les sommets d'un graphe maritime. Ces zones d'intérêt disposent de limites géographiques. Ces zones sont reliées par des arcs orientés pour former un graphe de zones (G_Z). La figure 1 présente le graphe de zones de la région de Brest. Un itinéraire peut ainsi être défini comme une suite ordonnée de zones reliées deux à deux par des arcs, correspondant à une chaîne du graphe de zones. Un itinéraire étant composé d'au moins un arc, il dispose par conséquent d'une zone de départ Z_D et d'arrivée Z_A .

2.3 Trajectoire suivant un itinéraire

Les trajectoires peuvent suivre ces itinéraires. Une trajectoire T_o est considérée comme suivant un itinéraire I passant par des sommets du graphe G_Z si :

- pour chaque zone de l'itinéraire I , il existe au moins une position P_{oj} de la trajectoire T_o dans cette zone respectant la relation d'ordre temporel de I ,
- aucune position P_{oj} de la trajectoire T_o n'appartient à une zone du graphe autre que celles de l'itinéraire,
- seule la première position P_{od} de la trajectoire appartient à la zone de départ de l'itinéraire Z_D ,
- de même, seule la dernière position P_{oa} de la trajectoire appartient à la dernière zone de l'itinéraire Z_A .

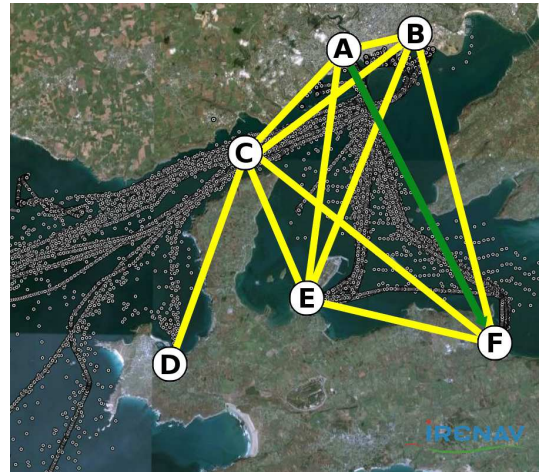


FIG. 1 – Graphe de zones de la région de Brest

3 Patrons spatio-temporels

3.1 Trajectoire médiane

Une fois les définitions de trajectoires et d'itinéraire posées, il est possible de construire la trajectoire médiane (Tr_{med}) des objets mobiles d'un certain type T suivant un même itinéraire I . Cette trajectoire médiane doit être déduite statistiquement à partir de l'ensemble des trajectoires répondant aux critères (itinéraire I et type T), nommé ci-après groupe homogène de trajectoires (GHT_{IT}). Afin de simplifier les phases de calcul appliquées à ce GHT_{IT} , chaque trajectoire du groupe est préalablement filtrée afin de ne conserver que les positions caractéristiques de la trajectoire ([2, 8]). Le calcul de la trajectoire médiane génère des nuages de positions interpolées, ordonnées selon un temps relatif ([8]). Le temps relatif est le temps écoulé depuis le départ de l'objet de la zone Z_D . En sélectionnant les valeurs médianes des latitudes et longitudes des différents nuages de positions, des positions médianes sont obtenues pour chaque temps relatif. En reliant ces positions médianes, la trajectoire médiane (T_{med}) de l'itinéraire est définie et sauvegardée dans la base de connaissances. Une approche statistique médiane a été préférée à une analyse basée sur la moyenne. En effet, cette approche est moins sensible aux écarts importants qui ne sont pas symétriques.

La figure 2 présente la trajectoire médiane (en jaune) du groupe homogène des trajectoires de "navires à passagers" effectuant l'itinéraire Brest - Lanvéoc.

3.2 Couloir spatio-temporel

Cette trajectoire médiane est ensuite complétée par un couloir spatio-temporel qui délimite l'espace géographique et temporel au sein duquel la majorité des trajectoires qui suivent cet itinéraire évoluent. Ce couloir spatio-temporel englobant les positions habituelles autour de la trajectoire médiane est également calculé par analyses statistiques. La première étude est spatiale. Elle s'appuie sur la droite per-

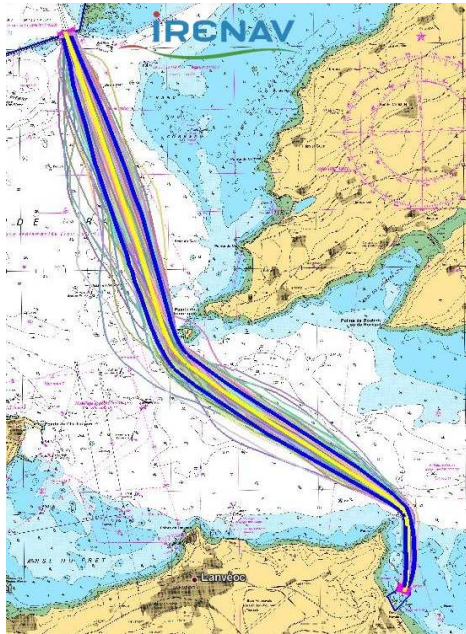


FIG. 2 – Trajectoire médiane et son couloir spatio-temporel

pendiculaire au cap (DPC) de chaque position de T_{med} . Le cap suivi par un navire représentant la direction suivie par l'objet mobile. Pour chaque trajectoire du GHT_{IT} , le point d'intersection avec la DPC est calculé, classé en deux parties (à droite ou à gauche) et ordonné par ordre de distance à la position de T_{med} . Pour chaque partie, le neuvième décile (seuil choisi empiriquement) de chaque côté de la trajectoire médiane est sélectionné. En reliant ces positions ordonnées, le couloir spatial de navigation est obtenu et sauvegardé dans la base de connaissances. La même analyse est effectuée en classant les positions obtenues par ordre de temps, ceci permettant d'obtenir les limites temporelles du couloir ([8]). La figure 2 présente le couloir spatial (en bleu) du groupe homogène des trajectoires de "navires à passagers" effectuant l'itinéraire Brest - Lanvéoc. L'ensemble trajectoire médiane associé au couloir spatio-temporel compose la route type.

4 Analyse temps réel

En s'appuyant sur cette base de connaissances, il est alors possible de détecter en temps différé ou en temps réel (temps réel souple) des situations inhabituelles et de qualifier le comportement d'un objet mobile.

4.1 Appariement à un itinéraire

Pour une nouvelle position P reçue, ce processus de détection des situations inhabituelles suppose préalablement d'associer cette position à un des patrons spatio-temporels liés à un arc d'un itinéraire du graphe de zones (figure 3.1). Dans cet objectif, sa trajectoire doit être appariée avec une des trajectoires médianes de la base de connaissances. Cet appariement s'effectue en fonction :

- de la position actuelle de l'objet mobile,
- du type de l'objet mobile,
- de l'historique de sa trajectoire,
- des intersections de sa trajectoire avec des zones du graphe,
- de l'éventuelle information concernant sa destination prévue.

Cet appariement est plus ou moins précis. En effet, si il existe plusieurs T_{med} proches de l'objet mobile et que sa destination finale est inconnue, l'appariement reste incertain. En revanche, si les destinations des objets mobiles sont connues (préalablement indiqué dans une feuille de route par exemple) ou bien si une unique T_{med} est proche de l'objet mobile, alors l'appariement est plus fiable. Si la destination est connue, à l'aide du graphe de zones, il est possible de déterminer la T_{med} à appairer à l'aide de la dernière zone traversée par la trajectoire. Par exemple, lorsqu'un navire à passagers quitte la zone de Brest (Zone A du graphe de la Figure 3.2) sa trajectoire peut être appariée à différentes routes types correspondant aux arcs du graphe de zones (A-E, A-C, A-B, A-F) représentées en marron sur la figure 3.2. Plus le navire se dirige vers sa destination finale, plus il est sensé suivre la trajectoire médiane associée à son itinéraire. Ainsi, comme indiqué sur la figure 3.3, la trajectoire médiane associée à la trajectoire du navire (en bleu clair) est sélectionnée en fonction de la position et de la trajectoire de l'objet mobile.

Une distance linéaire partielle est appropriée à cette tâche ([6]). L'objectif est de mesurer les distances linéaires entre la trajectoire T_r partant de la zone Z_D jusqu'à la position actuelle de l'objet mobile (représentée en bleu clair sur la figure 3.3) et les trajectoires médianes (T_{med}) (représentées en jaune sur la figure 3.3) associées aux différents arcs du graphe de zones partant de Z . Cette distance est calculée en utilisant les estampilles temporelles relatives des positions de la trajectoire. Effectivement, cette estampille temporelle permet de fixer des couples de positions homologues. Les positions relatives de T_r sont appariées avec les positions des T_{med} aux mêmes instants. De la sorte, il est possible de tenir compte des accélérations et des ralentissements des objets mobiles. La formule de la distance maximale linéaire partielle entre la trajectoire T_r et une trajectoire médiane T_{med} est donc :

$$Dist_{lp}(T_r, T_{med}) = \max_{(0 \leq t \leq d)} \|T_r(t) - T_{med}(t)\|$$

Cette distance est discrète, seules les positions relatives stockées de T_r et des T_{med} sont prises en compte. Pour chaque position entre l'instant de départ de la dernière zone ($t = 0$) et la durée depuis le départ ($t = d$) de la trajectoire T_r , la distance avec la position interpolée de T_{med} est calculée. La T_{med} la plus proche et à une distance linéaire partielle inférieure à un seuil prédéfini, est appariée avec la trajectoire T_r .

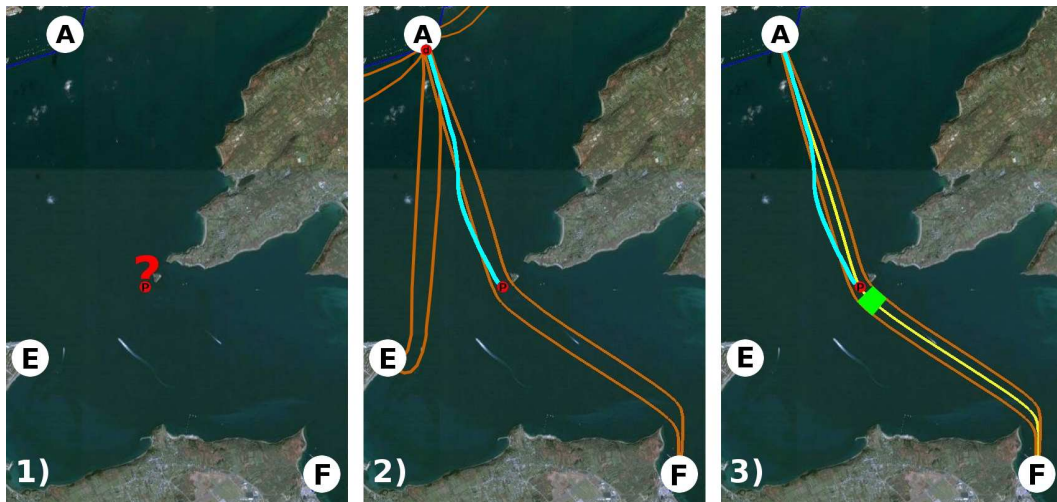


FIG. 3 – Appariement d’une trajectoire à une route type

4.2 Détection de comportement inhabituel

Une fois la T_{med} appariée à la trajectoire, la dernière position reçue de l’objet mobile peut être qualifiée à l’aide du couloir spatio-temporel associé pour ce type d’objet mobile. En fonction du temps relatif depuis la zone de départ de l’arc de l’itinéraire et du couloir spatio-temporel sélectionné, les 5 zones spatio-temporelles du couloir sont utilisées pour qualifier la position de l’objet (la zone de normalité est représentée visuellement sur les figures 3 et 4 par une zone verte).

Ainsi, en fonction de la position spatiale de l’objet mobile au temps relatif t , sa position peut être qualifiée comme étant :

- sur la route à l’heure (situation normale représentée par la zone verte)
- sur la route en retard (situation du point P sur les figures 3 et 4),
- sur la route en avance,
- à droite de la route,
- à gauche de la route.

5 Conclusion

Les travaux présentés s’intéressent à la problématique particulière du déplacement d’objets mobiles dans un espace ouvert. L’originalité de cette démarche d’analyse de similitude réside dans l’introduction de métaconnaissances (GHT_{IT} , trajectoire médiane, couloir spatio-temporel, route type) permettant de qualifier les comportements d’objets mobiles suivant un itinéraire dans un espace ouvert. Elle a été appliquée plus particulièrement au domaine maritime comme exposé dans notre exemple basé sur un jeu conséquent de données réelles (5 millions de positions stockées depuis 2007).

En ce qui concerne la surveillance du trafic maritime, les opérateurs peuvent avoir un grand nombre de navires à suivre. La transmission des informations qualitatives

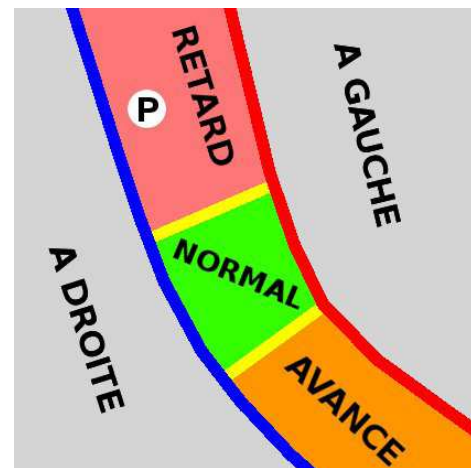


FIG. 4 – Qualification d’une position

concernant la trajectoire du navire leur permet de se focaliser sur un petit nombre de navires. De plus, ce module d’analyse est capable, une fois la trajectoire appariée, d’indiquer la destination du navire ou du prochain nœud du graphe ainsi que le temps nécessaire pour arriver à destination par comparaison avec les données de la trajectoire médiane.

La méthodologie présentée reste cependant générique. Elle pourrait par exemple être appliquée à l’étude du déplacement d’animaux ou de phénomènes naturels. Un prototype fonctionnel a été développé permettant le calcul et la sauvegarde dans une base de connaissances de la trajectoire médiane et du couloir spatio-temporel associé. La partie traitant de l’appariement en temps réel d’une trajectoire à un itinéraire a été modélisée, mais n’a cependant pas encore été implémentée dans ce prototype. Il est désormais important de valider cette dernière phase et de mettre à la disposition d’opérateurs de surveillance du trafic maritime l’ensemble du prototype.

Actuellement, Les trajectoires des objets mobiles sont découpées en s'appuyant sur le graphe de zones. Parallèlement, il est intéressant de scinder les trajectoires non plus selon leur itinéraire, mais selon des tronçons homogènes (ligne droite avec vitesse constante, virage pour changer de cap, etc.) afin de pouvoir décrire qualitativement les trajectoires d'objets mobiles et dans un deuxième temps de comparer qualitativement des trajectoires entre elles (ou avec la trajectoire médiane). L'objectif est donc de décrire les trajectoires à l'aide de variables linguistiques. Une description possible serait : *pour aller de la zone A à la zone B pour un objet de tel type, il faut aller tout droit avec une vitesse d'environ 20 km/h et un cap au sud-est, puis au bout de 10 minutes changer de cap pour aller à l'est et aller tout droit pendant 2 minutes.*

Finalement, une deuxième perspective à cette analyse serait de prendre en compte les spécificités dues à l'environnement dans lequel l'objet mobile évolue (les saisons et les marées dans le cadre de notre exemple). Il serait intéressant d'appliquer d'autres critères de sélection du GHT_{IT} ainsi que d'enrichir cette analyse en la couplant avec des travaux portant sur la prise en compte de l'environnement ([12]), des statistiques d'accidents ([5]) et des comportements illégitimes ([11]).

Références

- [1] C. C. AGGARWAL et P. S. YU : Outlier detection for high dimensional data. *SIGMOD*, 30(2):37–46, 2001.
- [2] F. BERTRAND, A. BOUJU, C. CLARAMUNT, T. DEVOGELE et C. RAY : *Web and Wireless Geographical Information Systems*, vol. 4857 de *Lecture Notes in Computer Science*, chap. Web Architecture for Monitoring and Visualizing Mobile Objects in Maritime Contexts, p. 94–105. Springer Berlin / Heidelberg, 2007.
- [3] D. BIRANT et A. KUT : Spatio-temporal outlier detection in large databases. *CIT. Journal of computing and information technology*, 14(4):291–297, 2006.
- [4] L. CHEN, M. T. OZSU et V. ORIA : Robust and fast similarity search for moving object trajectories. In *SIGMOD '05 : Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, p. 491–502, New York, NY, USA, 2005. ACM.
- [5] T. DEGRÉ : The use of risk concept to characterize and select high risk vessels for ship inspections. *12th International Congress of the International Maritime Association of the Mediterranean*, 6:37–49, 2007.
- [6] T. DEVOGELE : A new merging process for data integration based on the discrete fréchet distance. In *Advances in Spatial Data Handling : 10th International Symposium on Spatial Data Handling*, p. 167–181. Springer Verlag, 2002.
- [7] S. DODGE, R. WEIBEL et P. LAUBE : Phd showcase : Exploring movement–similarity analysis of moving objects. In *17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2009.
- [8] L. ETIENNE, T. DEVOGELE et A. BOUJU : Analyse de similarité de trajectoires d'objets mobiles suivant le même itinéraire : Application aux trajectoires de navires. *Ingénierie des Systèmes d'Information*, 14(5/2009):85–106, 2009.
- [9] J. LEE, J. HAN et X. LI : Trajectory outlier detection : A partition-and-detect framework. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on Data Engineering*, p. 140–149, 2008.
- [10] B. LIN et J. SU : Shapes based trajectory queries for moving objects. In *GIS '05 : Proceedings of the 13th annual ACM international workshop on Geographic information systems*, p. 21–30, New York, NY, USA, 2005. ACM.
- [11] M. MOREL, A. NAPOLI, A. LITTAYE, J. GEORGE, F. JANGAL, B. ALHADEF, C. SCAPEL, J. LEBREVELEC et D. DEJARDIN : Surveillance et contrôle des activités des navires en mer scanmaris. In *Workshop Interdisciplinaire sur la Sécurité Globale*, 2008.
- [12] A. RICHARD : L'optimisation des transits de navires marchands par l'utilisation des courants de marée. *Canadian Journal of Transportation*, 1, 2007.
- [13] Y. SAKURAI, M. YOSHIKAWA et C. FALOUTSOS : Ftw : fast similarity search under the time warping distance. In *PODS '05 : Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, p. 326–337, New York, NY, USA, 2005. ACM.
- [14] C.-B. SHIM et J.-W. CHANG : *Advances in Databases and Information Systems*, chap. Similar Subtrajectory Retrieval for Moving Objects in Spatio-temporal Databases, p. 308–322. Springer, 2003.

Représentation de l'espace et des déplacements dans la simulation de comportements humains ; application au trafic maritime

Thierry Le Pors^{1,2}

Thomas Devogele¹

Christine Chauvin²

¹ Institut de recherche de l'Ecole navale, BCRM Ecole navale, CC 600, 29 240 Brest cedex 9.
tél: + 33 (0)2 98 23 40 21. Fax: + 33 (0)2 98 23 38 57.

mailto: thierry.le_pors@ecole-navale.fr, thomas.devogele@ecole-navale.fr.

² Université de Bretagne Sud – CRPCC/LESTIC – Centre de recherche, rue de Saint-Maudé. 56321 Lorient Cedex.
tél : (33) 0297874521. Fax : (33) 0297874500.

mailto:christine.chauvin@univ-ubs.fr.

1 Introduction

Le travail présenté a pour objectif la simulation de la prise de décision d'experts, et de leur intégration au sein d'un simulateur multi-agents du trafic maritime. Le modèle utilise une représentation des connaissances à base de patrons pour prendre une décision. Dans ces patrons sont représentés l'espace et les déplacements à l'aide de variables linguistiques. Par exemple un objet peut être qualifié de rapide et proche. Zimmerman et Freksa proposent notamment une représentation générique de l'espace en fonction des orientations et des proximités (Zimmerman, Freksa, 1996). Pour intégrer dans des simulations à base de systèmes multi-agents ces notions qualitatives, l'utilisation de règles d'inférences classique n'est pas adaptée.

Dans le contexte maritime, le comportement d'expert été analysé par les travaux de (Chauvin, Lardjane, 2008), dans le cadre des évitements de collisions entre ferries et cargos dans la zone du Pas-de-Calais. Ces travaux identifient les différentes variables linguistiques utilisées par les marins et leurs valeurs sémantiques associées.

Ces variables linguistiques peuvent être rattachées à la notion de stimulus critique ou d'indice critique utilisé dans le paradigme de la prise de décision naturelle (Endsley, 1997).

Le simulateur multi-agents du trafic maritime CogTRANS (Le Pors et al, 2009) a permis d'intégrer ce système de prise de décision naturelle au sein de rôles. La notion de rôle se rapporte ici à un système organisationnel dans lequel les rôles joués par des agents sont décrits au sein de groupes. La notion de rôle permet de séparer les différentes fonctions d'un agent, et dans le cadre d'une simulation cognitive, d'identifier ses différents buts (Ferber, Gutknecht, 1998).

La première section de cet article présente donc le modèle de la prise de décision d'experts utilisant des variables linguistiques (**Fig. 1**). La seconde section indique comment sont manipulées les données numériques de la simulation pour permettre leur utilisation par des patrons utilisant des variables linguistiques. Enfin nous concluons par une présentation de nos résultats obtenus précédemment et par une réflexion sur le modèle.

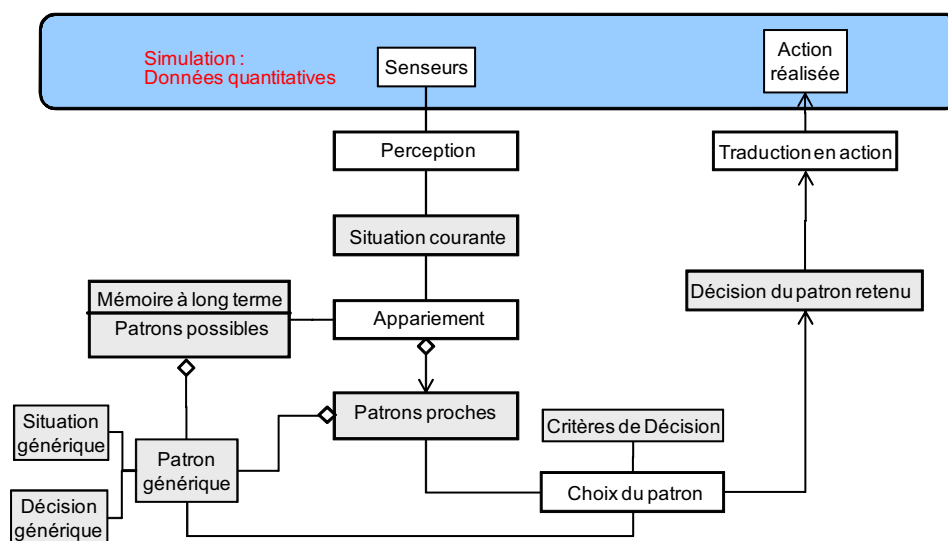


Fig. 1– système de prise de décision d'experts sous CogTRANS.

2 Modélisation cognitive des décisions d'experts

Le rôle de prise de décision développé est présenté sur la **Fig. 1**. Le but de ce système est de reproduire la prise de décision d'un expert en s'appuyant sur la théorie RPD (Recognition Primed Decision) (Klein, 1997) proche d'un système de reconnaissance à partir de cas et issue de la psychologie cognitive.

Le schéma montre bien la séparation des données numériques de la partie traitant des variables linguistiques. L'interface entre la prise de décision qualitative et ces données sont traitées par un module de perception en entrée et un module d'action en sortie. Cette prise de décision s'effectue ainsi en quatre étapes : une perception des données récupérées des différents capteurs de l'agent. Ces données sont transformées en données qualitatives en utilisant des fonctions floues (le système est décrit dans la partie suivante). Ensuite, une représentation mentale de la situation est comparée à un ensemble de données prototypiques, formant la base de connaissances des experts. Un algorithme de pattern-matching (appariement) va identifier le ou les patrons les plus proches de la situation à l'aide d'une distance sémantique, calculée entre chaque patron et la situation courante. Puis un choix doit être effectué entre ces différents patrons. Le choix qui a été privilégié ici est celui de l'utilisation de critères de décision lié à des profils d'agents. Dans notre exemple, les critères sont le respect du règlement, la sécurité de l'action et l'économie de temps. Enfin, la décision d'action liée au patron est transformée en données numériques. Dans le cadre du trafic maritime, la décision d'évitement est traduite sous la forme d'une modification de la vitesse ou d'un cap.

Il est important de noter que le système de prise de décision présenté ici est un composant de l'agent sous la forme de rôle. Il est intégré au sein d'un système organisationnel qui définit à quel moment les rôles doivent être joués. Par exemple, quand une décision d'action est choisie, cela déclenche le jeu du rôle « manœuvre d'évitement » puis cette manœuvre d'évitement est stoppée lorsque le navire est évité. La section suivante détaille le processus de traduction des données de simulation en valeurs linguistiques et la représentation des patrons.

3 Appariement de données numériques à des variables linguistiques

Les experts raisonnant à partir de variables linguistiques, les différentes données numériques se voient donc appariées à ces variables. Chaque variable linguistique est associée à un ensemble de fonctions floues (pour chacune des valeurs sémantiques possibles), (Zadeh, 1999). Une valeur inconnue permet de gérer l'incertitude liée à la perception.

En se basant sur des entretiens avec la population de marins et une étude statistique de leur comportement en mer, la fonction d'appartenance associée à la valeur linguistique de proximité au point de croisement a pu être définie (voir **Fig. 2**).

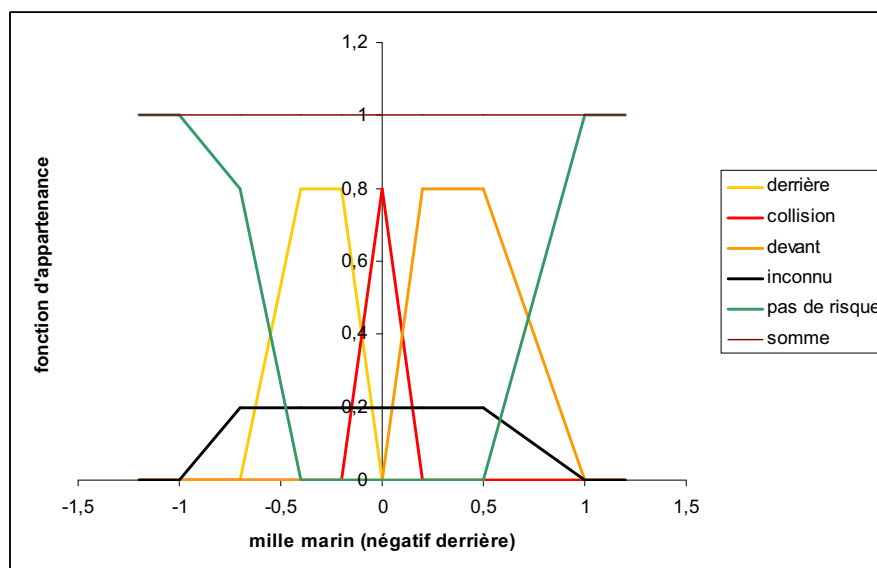


Fig. 2 – Fonction d'appartenance en fonction de la distance en milles marins pour les 5 sous-ensembles flous.

Par exemple, pour l'indice pertinent "position de croisement", les 5 valeurs possibles sont :

- 1 croisement par l'arrière : la distance lors du futur croisement (DCPA) est relativement faible, mais le ferry passe derrière, environ entre 0,4 et 0,2 mille marin (Un mille marin = 1,852 km),
- 2 collision : la distance est proche de 0 mille marin,
- 3 croisement par l'avant : le ferry passe devant, entre 0,2 et 0,5 mille marin,
- 4 inconnu,
- 5 pas de risque de collision (la distance jugée trop éloignée).

Une dissymétrie par rapport à l'axe des ordonnées est due au fait qu'il est plus dangereux de passer devant que derrière. Les risques de collision ne sont pas appréhendés par tous les navires de la même manière. Les distances seuils doivent donc être adaptées pour chaque type de navires.

Ces fonctions floues permettent la construction d'une situation courante qui doit ensuite être comparée aux patrons de la base de connaissance : un patron contient les mêmes informations que la situation courante mais se voit en plus associer une décision d'action prototypique. Par exemple, un ferry peut avoir deux choix possibles dans une situation donnée en fonction de la variable linguistique « proximité au point de croisement ». Pour une situation de croisement avec un petit cargo, à une distance lointaine, le ferry venant de la gauche, le règlement indique que le navire doit éviter celui venant de droite en effectuant une manœuvre à droite. Le ferry possède ici deux patrons sur lesquels s'appuyer :

- 1 un patron pour un point de croisement en collision avec l'autre navire qui privilégie de ne rien faire (le marin considère qu'il est trop loin pour effectuer une action et attend de rentrer dans le champ d'application de la règle).
- 2 un patron pour un point de croisement où le ferry tend à passer l'autre navire par l'avant (le marin considère qu'il est trop loin pour que son action entre dans le champ de la règle et qu'il peut donc entreprendre une manœuvre à gauche, moins coûteuse en temps qu'une manœuvre à droite).

4 Conclusion

L'intérêt de ce système a pu être prouvé lors de simulations présentées dans l'article (Le Pors et al., 2009). Il a été montré que 87% des situations observées dans le cas réel pouvaient être prédites par la simulation, composée d'agents utilisant ce type de rôles pour la prise de décision. Toutefois, les situations proposant un contexte géographique particulier ne sont pas simulées de façon correcte. Ces situations sont celles où le marin a

choisi sa manœuvre en fonction de sa position sur sa voie de navigation ou lorsqu'il a décidé d'éviter un « train de navires » (plusieurs navires à plus grande distance dont les trajectoires peuvent amener à de nouveaux cas d'évitement). Ce modèle pourrait être amélioré par l'ajout de nouvelles valeurs linguistiques (par exemple l'éloignement à la voie de navigation).

Cette approche et cette modélisation de l'espace et des déplacements est générique. Elles peuvent être employées pour toute simulation d'activité d'experts concernant des objets mobiles.

Bibliographie

- [1] Chauvin C. and Lardjane S., (2008). Decision-making and strategies in an interaction situation: collision avoidance at sea. *Transportation Research, part F*, 11 (4), p. 259-269.
- [2] Endsley M. R., (1997). The role of situation awareness in naturalistic decision making. In Zsombok, C. E. & G. Klein (eds.), *Naturalistic decision making*, p. 269-283.
- [3] Ferber J. and Gutknecht O., (1998). A meta-model for the analysis and design of organizations in multi-agents systems. In Demazeau Y, editor, *ICMAS'98*, Paris, p. 128-135.
- [4] Klein, G., (1997). The recognition-primed decision (RPD) model: looking back, looking forward. In Zsombok C.E., G. Klein (eds.), *Naturalistic Decision Making*, p. 285-292.
- [5] Le Pors T., Devogele T., Chauvin C. (2009) Multi-agent system integrating naturalistic decision roles: application to maritime traffic, In *Proceedings of the IADIS International Conference on Intelligent Systems and Agents*, IADIS Press, p 100-107.
- [6] Zadeh L.A., (1999). From Computing with Numbers to Computing with Words -- From Manipulation of Measurements to Manipulation of Perceptions, *IEEE Transactions on Circuits and Systems*, 45, p. 105-119.
- [7] Zimmerman K., Freksa C. (1996) Qualitative spatial reasoning using orientation, distance and path knowledge, *Applied Intelligence*, vol. 6, num. 1, p 49-58.

Fouille de paysages agricoles : analyse des voisinages des successions d'occupation du sol

Jean-François Mari¹

El Ghali Lazrak²

Marc Benoît²

¹ LORIA, UMR CNRS 7503 et INRIA-Grand Est

² INRA, UR 055, SAD ASTER

LORIA, UMR CNRS 7503 et INRIA-Grand Est

B.P. 239, F-54506 Vandœuvre-lès-Nancy

jfmari@loria.fr

INRA, UR 055, SAD ASTER

domaine du Joly, F-88500 Mirecourt

{lazrak, benoit}@mirecourt.inra.fr

Résumé

Nous présentons deux approches stochastiques à l'aide de modèles de Markov cachés (HMM) pour décrire les relations de voisinages entre successions d'occupation du sol dans un paysage agricole représenté par sa mosaïque de parcelles. Une première approche qualifiée de temporo-spatiale recherche à l'aide d'un HMM des classes de successions d'occupation du sol qui sont par la suite localisées. Ces dernières s'agglomèrent au sein de régions compactes (ou patches). Nous présentons une méthode de clustering spatial qui représente les patches à l'aide de deux distributions : une distribution qui représente l'homogénéité spatiale et l'autre, l'influence des voisinages à la lisière des patches. La deuxième approche analyse les dépendances spatio-temporelles représentées par le graphe du système de voisinages entre parcelles et enrichit les connaissances sur l'organisation territoriale de l'activité agricole en permettant l'identification de quartiers culturels au sein desquels les occupations des parcelles manifestent de forts degrés d'attraction.

Mots Clef

HMM2, classification spatiale, classification temporelle, successions de culture.

Abstract

We present two stochastic approaches using Hidden Markov Models to describe the relations between the neighborhoods of land use successions in a territory represented as a parcel mosaic. The first approach, qualified as time-spatial, clusters first the land use successions and, next, locates them in the territory. They appear agglomerated into compact areas called patches. We describe a clustering method that describes the patches by means of 2 distributions

of land use successions : one to model the spatial homogeneity and the other to model the influence of the neighborhood. The second approach analyzes the time spatial dependencies in the neighborhood graph and gives valuable informations in defining cultural districts in which the land use successions show a high degree of attractivity.

Keywords

HMM2, temporal clustering, spatial clustering, land use successions

1 Introduction

Le paysage agricole peut être perçu comme un assemblage de polygones de tailles différentes – les parcelles – où chaque parcelle porte une occupation du sol (OCS). L'ensemble de ces OCS est choisi par les agriculteurs.

La façon dont chaque agriculteur organise son territoire est un processus à la fois temporel et spatial. Les dépendances entre l'espace et le temps deviennent complexes à l'échelle des paysages agricoles où la mosaïque agricole est construite par de nombreux acteurs. Ceux-ci sont soumis à des opportunités ou des contraintes de nature différente agissant à diverses échelles. En allant du plus proche au plus lointain, on peut distinguer les contraintes / opportunités de voisinage et de localisation par rapport à l'eau ou aux villages, celles d'ordre pédologique et topographique, celles imposées par la présence d'une coopérative et enfin celles définies par la politique agricole à une échelle supra nationale. L'ensemble de ces contraintes / opportunités nous fait émettre l'hypothèse que l'activité des différents agriculteurs, sans être concertée, conduit à l'aménagement du paysage d'une manière convergente et intelligente qu'il nous faut comprendre avant de pouvoir la modifier.

Pour analyser les dépendances temporelles et spatiales

entre OCS, nous nous appuyons sur 2 hypothèses :

1. l'OCS d'une parcelle dépend de l'OCS des parcelles voisines (hypothèse de champs de Markov);
2. l'OCS d'une parcelle une année donnée dépend de l'OCS de cette parcelle les années précédentes (hypothèse de chaîne de Markov).

Dans cet article, nous présentons une analyse des voisinages des successions par une approche stochastique avec des HMM d'ordre 2 (HMM2). Nous commençons par présenter le territoire d'étude et le corpus spatio-temporel d'OCS. Ensuite, nous présentons deux approches complémentaires d'analyse des voisinages des successions d'OCS : (i) une approche qualifiée de temporo-spatiale où il s'agit de rechercher des classes de successions d'OCS qui sont par la suite localisées et fait émerger des patches; (ii) une deuxième approche qui envisage le couple (OCS, OCS d'une parcelle voisine) pour analyser les dépendances spatiales et temporelles entre OCS voisines en vue de valider et compléter les résultats de la première approche.

2 Représentation de la mosaïque parcellaire

2.1 Définitions

Le composant de base dans un territoire agricole est la parcelle : polygone de taille variable possédant un couvert : l'OCS. Elle est délimitée géographiquement par une route, un chemin, ... ou artificiellement par l'agriculteur qui subdivise le territoire de son exploitation pour respecter un assolement : ensemble des surfaces allouées à chaque culture. Les frontières des parcelles changent chaque année en fonction des choix des agriculteurs. Pour tenir compte de ce changement, les enquêteurs définissent l'ensemble des micro-parcelles comme étant l'union de toutes les frontières des parcelles pendant la période d'étude.

Le paysage agricole étudié s'étend sur 350 km² dans la Plaine de Niort. Il est enquêté depuis plus de 12 ans pour la localisation et les occupations de ses parcelles. Il y a environ 20000 micro-parcelles. Une micro-parcelle n'a hébergé qu'une succession de culture pendant la période d'étude. Les relevés d'OCS issus des enquêtes annuelles sont stockés dans un système d'information géographique sous format vectoriel et constituent une couche d'informations temporelles et spatiales. Cette couche d'information vectorielle a été rasterisée avec une grille de points régulièrement espacés (10m x 10m). Le corpus résultant est une matrice où les colonnes représentent les OCS année par année et les lignes, les différents points d'échantillonnage localisés. Le corpus compte au total 47 OCS que nous avons regroupées, dans un travail antérieur [2] (Tab. 1) suivant une démarche tenant compte de la similitude des conduites culturales. Nous considérons plus particulièrement 5 OCS appartenant à deux classes contrastées de couverts :

- les couverts dynamiques : Blé, Tournesol, Colza ;
- et les couverts pérennes : l'Urbain et les Prairies.

OCS	Fréq. cumulée
Blé (B)	0.337
Tournesol (T)	0.476
Colza (C)	0.600
Urbain (U)	0.696
Prairies et Luzernes (P)	0.774
Maïs (M)	0.850
Forêts et friches (F)	0.884
Orge d'hiver (O)	0.918
Ray-grass (Y)	0.942
Pois (S)	0.964
Autres (A)	1.000

TAB. 1 – Composition et fréquences moyennes sur le territoire des OCS adoptées

2.2 Choix de l'observation élémentaire

Nous envisageons plusieurs observations élémentaires qui constitueront les modalités d'un pixel de l'image représentant la mosaïque agricole :

1. l'OCS en un point d'une parcelle représentant son occupation ;
2. la succession d'OCS de longueur fixe (par exemple T-B-C-B) représentant la suite temporelle (4 ans dans notre exemple) des OCS en un point d'une parcelle ;
3. le couple (OCS, OCS d'une parcelle voisine).

La première observation est utile pour retrouver les rotations dominantes selon la méthode de fouille développée par [3, 5].

L'utilisation de successions d'OCS de longueur fixe permet de différencier les rotations de cultures sur des périodes différentes : par exemple les rotations bi, tri ou quadri-annuelles. Plus la succession est longue, plus elle intéresse l'agronome, mais plus le nombre d'observations sera important. Ce dernier conditionne la taille des distributions des états du HMM.

Enfin, l'utilisation de couples (OCS, OCS d'une parcelle voisine) permet de fouiller les voisinages entre cultures et leurs évolutions. Ceci permet de quantifier l'attraction spatiale des OCS comme le révèle la méthode décrite au §5.

2.3 Choix de la longueur de succession et de la résolution spatiale

L'exécution des algorithmes de classification dépend de deux facteurs : (i) la taille de la matrice des données à travers la résolution d'échantillonnage spatial qui joue sur le temps de calcul, (ii) et la longueur de la succession d'occupation du sol qui joue sur la place occupée par la représentation des distributions de probabilité. Afin de disposer d'un critère objectif pour le choix de la résolution spatiale, la perte d'information en termes de diversité des successions d'OCS a été quantifiée pour différentes résolutions d'échantillonnage spatial (figure 1-a). La résolution de 80 m x 80 m a été retenue. Cette résolution a permis d'obtenir une matrice de données 64 fois plus réduite que la ma-

trice de données initiale avec une perte de seulement 6% en termes de nombre de successions.

Nous nous sommes intéressés ensuite à l'influence de la longueur des successions sur l'entropie du système. Nous avons comparé le nombre de successions différentes à celles produites aléatoirement dans le même territoire. Chaque OCS est tirée selon une loi uniforme sans tenir compte ni de sa localisation ni des OCS précédentes au même endroit. La figure 1-b montre qu'à partir d'une longueur de succession de 4 années, la zone d'étude commence à se distinguer nettement du modèle où les successions sont choisies de manière aléatoire dans la zone d'étude. Ceci justifie notre choix des successions de 4 ans (appelées par la suite quadruplets) comme une observation élémentaire pertinente.

2.4 Échantillonnage spatial à résolution variable

Dans la mosaïque parcellaire, le système de voisinage est irrégulier. Une parcelle a un nombre quelconque de parcelles avec lesquelles elle partage une frontière commune. Nous modélisons la mosaïque parcellaire par un champ de Markov irrégulier dans lequel la parcelle s'inscrit dans une succession temporelle d'OCS. L'ensemble constitue une image de successions plutôt qu'une succession d'images et ne peut être étudié directement par les modèles numériques d'images appliqués au suivi de trajectoires d'objets [6]. Enfin, les territoires étudiés ne sont pas carrés et les parcelles ajoutées pour leur donner une forme carrée doivent être en nombre minimum afin de ne pas perturber la classification spatiale.

Pour transformer cette représentation 2D en représentation 1D afin d'utiliser des HMM plutôt que des champs de Markov, nous cherchons un parcours du graphe des voisinages. Comme ce problème est NP-difficile, nous avons adopté une solution approchée en définissant un parcours fractal sur la grille régulière et en adaptant la profondeur de fractalisation à la taille de la micro-parcelle. Un processus récursif fusionne les points situés sur la courbe quand ils correspondent au motif générateur et qu'ils sont dans la même micro-parcelle et donne naissance à une courbe fractale à profondeur variable appelée dans la suite *courbe fractale réduite* (cf. Fig. 2). Ce balayage particulier introduit une normalisation des surfaces des parcelles ; les grandes parcelles sont sous représentées alors que les points au voisinage des frontières entre parcelles sont sur représentés.

3 Classification temporo-spatiale des successions

3.1 Classification temporelle des successions

Nous envisageons un HMM2 dont la topologie est donnée Fig.3 selon la méthode de fouille de données temporelles présentée dans [2]. Les successions d'OCS de chaque micro-parcelle sont utilisées dans l'algorithme Forward-Backward pour l'estimation du HMM2 [4] sans tenir compte

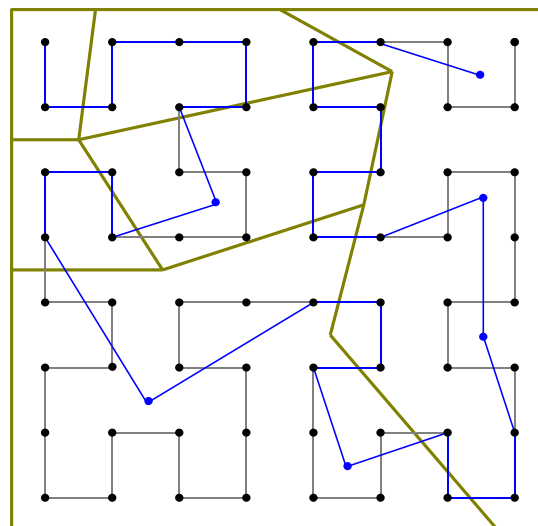


FIG. 2 – Prise en compte de la taille des parcelles dans le parcours du plan par la courbe fractale de Hilbert-Peano. L'exemple montre deux fusions successives dans la parcelle située en bas à gauche aboutissant à l'agglomération de 16 points en un seul

de leur localisation. Les distributions initiales et finales des états sont données Tab. 2.

Les états du HMM2 ont été initialisés à l'aide de distributions uniformes de quadruplets d'OCS correspondant aux motifs de recherche décrits dans le tableau 2-a. Le HMM2 utilisé possède autant d'états que de $S(X)$ et il est appris sur le corpus.

Dans le territoire d'étude, le Blé (B), le Tournesol (T) et le Colza (C) sont le plus souvent intégrés dans une même succession de 4 ans (p. ex. T-B-C-B, T-B-T-B, C-B-C-B, T-B-B-B, C-B-B-B). La classification avec une classe commune à ces trois cultures a permis d'obtenir des résultats plus cohérents qu'avec les classes séparées. Cette classe, notée $S(B,T,C)$, regroupe les quadruplets impliquant au moins l'une de ces trois cultures. Avec ce regroupement, nous distinguerons par la suite 3 classes de successions : $S(B,T,C)$, $S(U)$ et $S(P)$. Le tableau 2 résume la distribution des observations à l'intérieur de ces classes de successions après apprentissage du corpus d'étude sans prise en compte de la composante spatiale.

de leur localisation. Les distributions initiales et finales des états sont données Tab. 2.

4 Spatialisation des $S(X)$

Les localisations des classes de successions d'OCS ont été recherchées à l'aide de modèles de Markov cachés hiérarchiques d'ordre 2 (HHMM2) [4]. Le HHMM2 utilisé est un modèle ergodique qui comporte un état spatial par classe

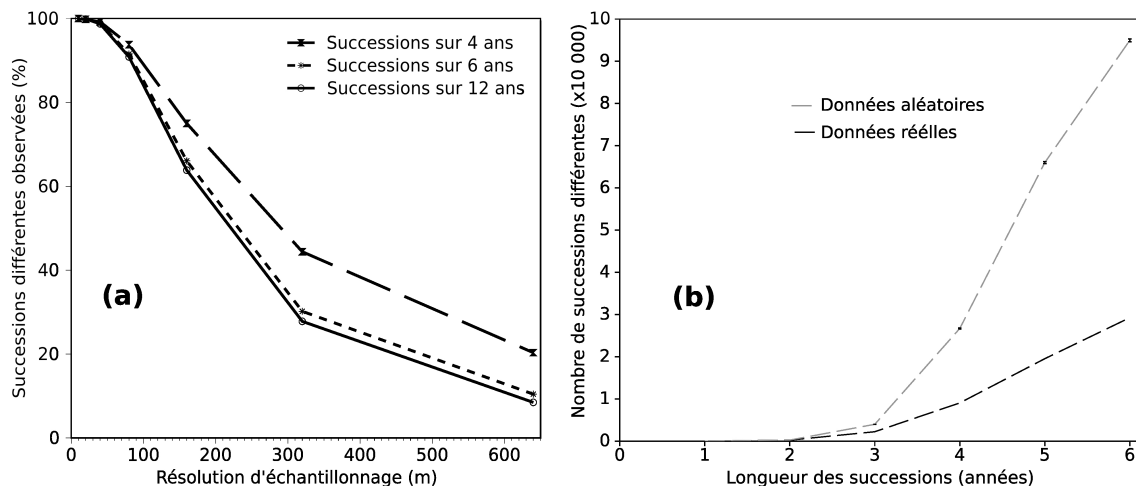


FIG. 1 – (a) Évolution de la perte d'information spatiale en fonction de la résolution d'échantillonnage. (b) Comparaison de la diversité des successions d'OCS entre les données réelles et des données générées aléatoirement pour différentes longueurs de successions

(a)

		S(X)		
t	t+1	t+2	t+3	
X	?	?	?	
?	X	?	?	
?	?	X	?	
?	?	?	X	

(b)

S(B,T,C)	S(U)	S(P)	
T-B-C-B	0.128	U-U-U-U 0.977	P-P-P-P 0.377
C-B-T-B	0.112	B-U-U-U 0.002	P-P-P-B 0.040
B-T-B-C	0.112	U-U-U-B 0.002	B-P-P-P 0.026
B-C-B-T	0.105	P-U-U-U 0.002	Y-P-P-P 0.026
T-B-T-B	0.072	T-B-U-U 0.001	T-P-P-P 0.025
C-B-C-B	0.065	U-U-B-U 0.001	B-T-P-P 0.018
B-C-B-C	0.062	U-U-B-U 0.001	P-P-P-T 0.016
B-T-B-T	0.059	T-U-U-U 0.001	P-P-T-B 0.015
B-C-B-B	0.033	P-P-U-U 0.001	A-P-P-P 0.015
B-B-C-B	0.024	U-U-U-P 0.001	P-P-B-C 0.013

TAB. 2 – Description des états du HMM2 utilisé pour la classification temporelle avant et après l'apprentissage. En (a) sont représentés les gabarits des quadruplets équiprobables impliquant l'OCS X. ? désigne une quelconque OCS. En (b) sont données les distributions des quadruplets dans les classes à l'issue de l'apprentissage sans prise en compte de l'espace

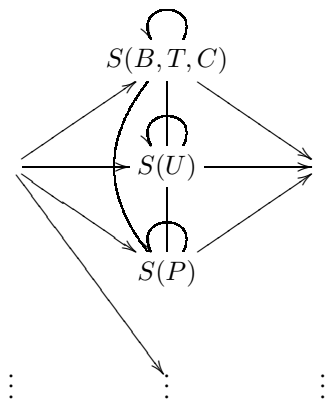


FIG. 3 – HMM2 pour la classification temporelle des successions. Les successions d'OCS de chaque micro-parcelle estiment les états S(X) sans utiliser l'information de localisation. Les successions d'OCS sont représentées par des quadruplets d'OCS se chevauchant. Tous les états du HMM2 ne sont pas représentés

de successions à localiser comme le montre la figure 4. Chaque état spatial est défini par un HMM2 ergodique à 2 états : (i) un état de gabarit représentant la classe S(X) et initialisé par un apprentissage temporel tel qu'il est décrit dans le §3.1 et (ii) un état de réserve destiné à capturer les successions au voisinage ne répondant pas aux critères spécifiés dans l'état de gabarit. Cet état de réserve est appelé par la suite, état de voisinage. Le tableau 3 présente les classes de successions S(X) spatialisées avec leurs états de voisinages respectifs. Le fait d'analyser les successions dans l'ordre imposé par la courbe de Peano réduite (cf. Fig. 2) introduit un changement dans la distribution des quadruplets. Dans chaque état du HMM hiérarchique, l'état équiprobable capte les exceptions dues aux effets de voisinage. Un site n'est plus classé seulement en fonction de ses caractéristiques temporelles mais subit l'influence de ses voi-

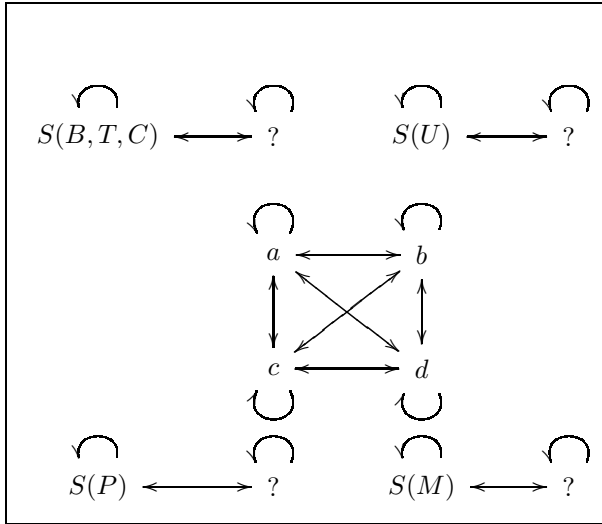


FIG. 4 – Chaque état a, b, c, d du HHMM2 maître est un HMM2 temporel dont les états sont $s(x)$ et un état équiprobable noté $?$ [4]. Tous les états du HHMM2 ne sont pas représentés

sins. La classe $S(B,T,C)$ spatialisée montre une distribution de fréquences légèrement différente par rapport à la distribution de la même classe non spatialisée (Tab. 2). L'ordre des quadruplets a néanmoins été conservé dans les deux cas. L'état de voisinage accompagnant la classe $S(B,T,C)$ spatialisée indique que les patches caractérisés par des successions impliquant majoritairement le Blé, le Tournesol et le Colza, comportent aussi des Prairies et luzernes (P-P-P) à 2% et des zones Urbaines (U-U-U-U) à 1%. La spatialisée des deux classes $S(U)$ et $S(P)$ a également modifié leurs distributions. L'état de voisinage de la classe Urbaine montre que les Prairies et luzernes sont les voisins les plus fréquents de l'Urbain. Inversement, l'Urbain est un voisin fréquent des Prairies et luzernes, mais les voisins les plus fréquents (13%) sont des successions du type $S(B,T,C)$ et plus particulièrement la rotation biennale : Blé-Tournesol qui représente, à elle seule, 5% du voisinage des Prairies et luzernes dans cette classe de patches.

5 Validation par étude des cliques

Pour mieux interpréter le contenu de l'état de réserve dans les HMM2 définissant les états du HHMM2 utilisé dans la classification spatiale, nous avons fouillé le graphe des voisinages des parcelles en étudiant l'évolution temporelle des couples (OCS d'une parcelle, OCS de la parcelle voisine) comme mentionné §2.4. Le parcours du graphe des voisinages est représenté par la courbe fractale réduite comme le font [1, 7] dans le cas d'un système de voisinage régulier. Une clique est un ensemble de parcelles deux à deux voisines, c'est à dire ayant une frontière commune. Une configuration des OCS à l'instant t d'une clique sera abusivement appelée "clique temporelle". Dans cet article, nous considérons des cliques simples faites de deux parcelles

$S(B,T,C)$	État de voisinage		$S(U)$	État de voisinage			
T-B-C-B	0.08	P-P-P-P	0.02	U-U-U-U	0.60	P-P-P-P	0.03
C-B-T-B	0.07	B-B-C-B	0.01	?-U-U-U	0.01	?-?-?-?	0.03
B-T-B-C	0.07	B-B-T-B	0.01	?-?-U-U	0.01	T-B-C-B	0.01
B-C-B-T	0.07	B-C-B-B	0.01	?-?-?-U	0.01	B-T-B-C	0.01
T-B-T-B	0.04	U-U-U-U	0.01	?-?-?-?	0.00	C-B-T-B	0.01
C-B-C-B	0.04	F-F-F-F	0.01	P-U-U-U	0.00	T-B-T-B	0.01
B-C-B-C	0.04	B-T-B-C	0.01	U-U-U-B	0.00	B-C-B-T	0.01
B-T-B-T	0.03	B-T-B-B	0.01	B-U-U-U	0.00	B-U-U-U	0.01
B-C-B-B	0.02	A-A-A-A	0.01	?-P-U-U	0.00	U-B-U-U	0.01
B-B-C-B	0.02	B-C-B-T	0.01	T-U-U-U	0.00	B-T-B-T	0.00
			S(P)		État de voisinage		
			P-P-P-P	0.20	T-B-T-B	0.03	
			Y-P-P-P	0.01	B-T-B-T	0.02	
			B-P-P-P	0.01	T-B-C-B	0.02	
			P-P-P-B	0.01	B-T-B-C	0.01	
			?-P-P-P	0.01	C-B-T-B	0.01	
			P-P-P-T	0.01	B-C-B-T	0.01	
			B-Y-P-P	0.01	B-T-B-B	0.01	
			T-P-P-P	0.00	U-U-U-U	0.01	
			P-P-T-B	0.00	B-B-T-B	0.01	
			T-B-P-P	0.00	C-B-C-B	0.01	

TAB. 3 – Distributions des quadruplets dans les classes à l'issue d'une classification temporo-spatiale du corpus. ? désigne l'information manquante dans le corpus correspondant à des parcelles non enquêtées certaines années

voisines représentées par leur centre de gravité. Des résultats préliminaires nous ont montré que la distribution des OCS est isotrope : la direction définie par les 2 centres de gravité ne porte pas d'information.

Deux sites successifs (s_{l-1}, s_l) , $1 \leq l < L$ dans la courbe fractale réduite de longueur L définissent une clique. Sur chaque site s_l de la courbe fractale réduite, nous définissons un vecteur de traits o_l^t avec les OCS situées sur les cliques (s_{l-1}, s_l) :

$$o_l^t = ((s_{l-1}^t, s_l^t), (s_{l-1}^{t+1}, s_l^{t+1})), \quad (1)$$

$$0 \leq t < T - 1, \quad 1 \leq l < L$$

s_l^t étant l'OCS au temps t et à l'index l dans la courbe fractale réduite. t représente le temps sur la période d'étude (12 ans) et l l'index spatial dans le parcours du plan.

Au temps t , un paysage est représenté par une suite de $L - 1$ cliques temporelles se chevauchant. Nous considérons aussi $T - 1$ représentations de paysage pour couvrir les T années de la période d'étude.

Les sites successifs sur la courbe fractale réduite résidant dans la même parcelle ne nous intéressent pas dans l'étude des voisinages entre parcelles. Les vecteurs de traits satisfaisant les propriétés $(s_{l-1}^t = s_l^t)$ et $(s_{l-1}^{t+1} = s_l^{t+1})$ seront supprimés dans les distributions.

Le vecteur de traits o_l^t est la réalisation de 4 variables aléatoires S_t, N_t, S_{t+1} et N_{t+1} qui définissent le processus stochastique observable. Ce vecteur se compare aux quadruplets temporels définis dans le §3.1 où les 4 OCS représentaient les 4 occupations successives d'une parcelle. A présent, ce vecteur représente 2 OCS successives sur 2 parcelles voisines.

La modélisation stochastique à l'aide d'un HMM2 permet d'extraire des patches caractérisés par des distributions de cliques temporelles (cf. Tab. 4).

– L'analyse des colonnes S_t et S_{t+1} montre les dépendances temporelles entre OCS sur un site alors que l'analyse des colonnes N_t et N_{t+1} montrent les mêmes dépendances au niveau des parcelles voisines qui suivent

- donc la même logique de mise en valeur ;
- de même, l’analyse des colonnes S_t et N_t montre le degré d’attraction entre OCS ;
 - enfin, l’analyse globale permet de quantifier l’attraction des successions d’OCS.

La table 4-a montre que les prairies sont les voisins “préférés” des zones urbaines comme le montrait déjà le traitement décrit au §4. Ce résultat nous permet de penser que l’état de réserve de chaque HMM2 (cf. §4) capture les successions à la lisière des patches. Le tableau montre aussi que ces 2 OCS sont stables dans le temps.

La table 4-b montre une nouvelle régularité que l’approche temporo-spatiale n’avait pas révélé. Les patches contenant les rotations incluant le tournesol (T), le blé (B) et le colza (C) dans des rotations bi ou quadri-annuelles sont constitués de parcelles dans lesquelles l’OCS voisine sera l’OCS suivante. Ceci montre que l’organisation dans le temps de la mosaïque agricole implique une organisation dans l’espace. Ceci nous conduit à envisager la description de la mosaïque en terme de quartiers culturels formés de parcelles dans lesquelles les mêmes rotations sont pratiquées afin de réduire la variabilité temporelle et spatiale de cette mosaïque.

(a)

Freq.	S_t	N_t	S_{t+1}	N_{t+1}
0.014	U	P	U	P
0.014	P	U	P	U
0.007	T	U	B	U
0.007	U	T	U	B
0.007	U	B	U	C

(b)

Freq.	S_t	N_t	S_{t+1}	N_{t+1}
0.014	B	C	C	B
0.014	C	B	B	C
0.011	T	C	B	B
0.011	C	T	B	B
0.010	B	B	C	T
0.010	B	B	T	C
0.010	T	B	B	C
0.009	B	T	C	B
0.009	C	B	B	T
0.009	B	C	T	B

TAB. 4 – Distribution des cliques temporelles dans les patches urbains (a) et dans les patches contenant les successions à base de tournesol, blé et colza (b)

6 Discussion et conclusions

Dans cet article, nous avons développé deux approches visant chacune à décrire les relations de voisinages entre successions d’OCS à l’échelle du paysage agricole. Nous

utilisons des modèles de Markov cachés pour modéliser la variabilité temporelle et spatiale du territoire agricole. Une première approche qualifiée de temporo-spatiale recherche des classes temporelles de successions d’OCS qui sont par la suite localisées après ajout d’un état de voisinage qui capture les successions à la lisière des patches ; la deuxième approche consiste à analyser les dépendances spatio-temporelles des successions d’occupation du sol en vue de valider et d’enrichir les connaissances extraites par la première approche.

Les modèles de classification mis au point dans ce travail visent à améliorer l’état des connaissances sur l’organisation territoriale de l’activité agricole. Ces modèles permettent de simplifier la complexité apparente de l’organisation et de l’évolution du paysage agricole à travers l’identification de zones homogènes en termes de successions d’occupation du sol mais aussi à travers l’identification de quartiers culturels au sein desquels les occupations des parcelles voisines manifestent de forts degrés d’attraction. Chaque patche est décrit par deux distributions de successions : l’une décrivant son homogénéité spatiale, l’autre l’influence des patches voisins. Ceci nous permet de localiser la lisière du patche et permet une fouille plus détaillée, notamment une recherche de successions singulières ou innovantes.

A terme, ces techniques de classification développées et implémentées dans la boîte à outils ARPENTAGE sont à même de contribuer à améliorer les modèles de simulation des paysages visant à évaluer des risques divers dans des territoires agricoles à enjeux environnementaux comme les risques liés à la pollution de l’eau, à la diffusion des transgènes ou à la préservation d’espèces animales.

Remerciements

Nous remercions la région Lorraine, l’ANR BiodivAgrim, et l’API Ecogir pour leurs financements.

Références

- [1] B. Benmiloud and W. Pieczynski. Estimation des paramètres dans les chaînes de Markov cachés et segmentation d’images. *Traitement du signal*, 12(5) :433 – 454, 1995.
- [2] E.G. Lazrak, J.-F. Mari, and M. Benoît. Landscape regularity modelling for environmental challenges in agriculture. *Landscape Ecology*, Sept. 2009. <http://hal.inria.fr/inria-00419952/en/>.
- [3] F. Le Ber, M. Benoît, C. Schott, J.-F. Mari, and C. Mignolet. Studying Crop Sequences With CarrotAge, a HMM-Based Data Mining Software. *Ecological Modelling*, 191(1) :170 – 185, Jan 2006. <http://hal.archives-ouvertes.fr/hal-00017169/fr/>.
- [4] J.-f. Mari and C. Largouet. *Raisonnements sur l’espace et le temps : des modèles aux applications*, chapitre 9 et 12, pages 249 – 271 et 305 – 316. ISBN : 2-

7462-1455-5. Lavoisier, F. Le Ber edition, mars 2007.
<http://hal.archives-ouvertes.fr/hal-00126022/fr/>.

- [5] J.-F. Mari and F. Le Ber. Temporal and Spatial Data Mining with Second-Order Hidden Markov Models. *Soft Computing*, 10(5) :406 – 414, March 2006.
<http://hal.inria.fr/inria-00000197>.
- [6] H. Memin and P. Perez. *Problèmes inverses en imagerie et vision*, chapter Estimation du mouvement, pages 205 – 267. Lavoisier, A. Mohammad-Djafari edition, 2009.
- [7] W. Pieczynski. Markov models in image processing. *Traitement du signal*, 20(3) :255–278, 2003.