



HAL
open science

Actes des 17es Rencontres des Jeunes Chercheurs en Intelligence Artificielle

Maxime Lefrançois

► **To cite this version:**

Maxime Lefrançois. Actes des 17es Rencontres des Jeunes Chercheurs en Intelligence Artificielle : RJCIA 2019. Plate-Forme Intelligence Artificielle, Association Française pour l'Intelligence Artificielle, 2019. hal-02167252

HAL Id: hal-02167252

<https://hal.science/hal-02167252>

Submitted on 6 May 2024

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

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



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License



AfIA

Association française
pour l'Intelligence Artificielle

RJCIA

*Rencontres des Jeunes Chercheurs
en Intelligence Artificielle*

PFIA 2019



Table des matières

Maxime Lefrançois. Éditorial	4
Maxime Lefrançois. Comité de programme	6
I - PRÉSENTATION INVITÉE	7
C. Trojahn. How to write a good state of the art : should it be the first step of your thesis?	8
II - ARTICLES LONGS	9
Alix Goudyme, Tiago de Lima, Nathalie Chetcuti-Sperandio and Sylvain Lagrue. Intention et logique épistémique dynamique	10
Emmanuel Dimitry Ngounou Ntougam, Vivient Corneille Kamla, Yazan Mualla, Igor Haman Tchappi, Stéphane Galland, Jean Claude Kamgang and Yves Sébastien Emvudu Wono. Towards a multi-agent model to prevent damage caused by cocoa mirids to cocoa pods	19
Régis Pierrard, Jean-Philippe Poli and Céline Hudelot. Apprentissage de relations floues pour l'annotation sémantique expliquée avec peu de données	27
Matthieu Riou, Bassam Jabaian, Stéphane Huet and Fabrice Lefèvre. Renforcement en-ligne pour l'apprentissage conjoint de l'analyseur sémantique et du gestionnaire de dialogue d'un système d'interaction vocale	36
Iago Felipe Trentin, Olivier Boissier and Fano Ramparany. Insights about user-centric contextual online adaptation of coordinated multi-agent systems in smart homes	44
Laetitia Zaleski, Isabelle Alvarez, Jean-Pierre Briot and Martha Irving. Viaduc : Un assistant de viabilité pour la négociation dans un éco-socio-système	52
III - ARTICLES COURTS	63
Raphaël Gazzotti, Elisa Noual, Catherine Faron Zucker, Fabien Gandon, Alain Giboin, Virginie Lacroix-Hugues and David Darmon. Interface d'aide à la décision pour prédire l'hospitalisation de patients et planifier les actions préventives pour prévenir cet événement	64
Beatrice Napolitano, Olivier Cailloux and Paolo Viappiani. Simultaneous elicitation of committee and voters' preferences	69
Romain Orhand, Anne Jeannin-Girardon, Pierre Parrend and Pierre Collet. DeepExpert : vers une Intelligence Artificielle autonome et explicable	73
Guillaume Prévost, Stéphane Cardon, Tristan Cazenave and Éric Jacopin. Planification dans les jeux-vidéos : À quoi servent les coûts des actions?	76
Edouard Villain, Federico Nemmi, Xavier Franceries, Patrice Péran and Marie-Véronique Lelann. Diagnostic de l'atrophie multi-systématisée par l'utilisation de réseaux de neurones convolutifs multi-modaux à 3 dimensions	79
Laetitia Zaleski. Démonstration de l'assistant de viabilité Viaduc pour la gestion participative dans un éco-socio-système	82

Éditorial

Les [17e Rencontres des Jeunes Chercheurs en Intelligence Artificielle 2019](#) ont eu lieu du 2 au 4 juillet 2019 à Toulouse sur la [Plate-Forme Intelligence Artificielle \(PFIA 2019\)](#).

La Plate-Forme IA constitue un point de rencontre unique et convivial pour la communauté IA permettant de rapprocher les différentes disciplines qui la composent et d'établir des passerelles entre elles. À cette intention, cette plate-forme s'adresse à l'ensemble de la communauté francophone d'Intelligence Artificielle afin qu'elle aborde des problématiques communes. Elle accueillait cette année :

- la 5ème Conférence sur les Applications Pratiques de l'Intelligence Artificielle (CPIA),
- la 21ème Conférence sur l'Apprentissage Automatique (CAp),
- la 22ème Conférence Nationale en Intelligence Artificielle (CNIA),
- la 30èmes Journées Francophones Ingénierie des Connaissances (IC),
- la 14èmes Journées Planification, Décision et Apprentissage (JFPDA),
- la 27èmes Journées Francophones sur les Systèmes Multi-Agents (JFSMA),
- la 13èmes Journées d'Intelligence Artificielle Fondamentale (JIAF),
- la 17ème Rencontre des Jeunes Chercheurs en Intelligence Artificielle (RJCIA),
- la 26ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN – RECITAL),
- la Compétition hébergée DriveToGaether,
- l'évènement France@International.

Les Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA) sont destinées aux jeunes chercheurs en Intelligence Artificielle : doctorants, titulaires d'un doctorat depuis moins d'un an, ou étudiants de master supervisés pour un travail de recherche en vue de débiter un doctorat. L'objectif de cette manifestation est double :

- permettre aux jeunes chercheurs préparant une thèse en Intelligence Artificielle, ou l'ayant soutenue depuis peu, de se rencontrer et de présenter leurs travaux, et ainsi de former des contacts avec d'autres jeunes chercheurs et d'élargir leurs perspectives en échangeant avec des spécialistes d'autres domaines de l'Intelligence Artificielle ;
- former les jeunes chercheurs à la préparation d'un article, à sa révision pour tenir compte des observations du comité de programme, et à sa présentation devant un auditoire de spécialistes, leur permettant ainsi d'obtenir des retours de chercheurs de leur domaine ou de domaines connexes.

Le comité de programme cette année à été renouvelé avec un vingtaine de jeunes chercheurs couvrant les thématiques de la conférence. Les membres du comité de programme de CNIA et RJCIA des années précédentes ont été sollicités pour recommander des jeune chercheurs : (1) qui ont soutenu leur thèse entre 2014 et 2017, et (2) qui étaient à ce moment là en Post-Doctorat en France ou à l'étranger. 31 jeunes chercheurs ont été recommandés et donc sollicités à l'issue de cette phase. 29 d'entre eux ont acceptés de faire partie du comité de programme, et se sont chacun vus attribués un article long, et un article court. 27 ont rédigé leurs deux évaluations. Je tiens à remercier personnellement Jonathan Bonnet et Hugo Gilbert pour avoir fourni des évaluations supplémentaires de dernière minute.

Un des objectifs de RJCIA est de former les jeunes chercheurs à la préparation d'un article, et à sa révision pour tenir compte des observations du comité de programme. Les évaluateurs ont donc soumis des évaluations justes et qui ont permis au jeune chercheur de s'améliorer dans la préparation d'un article et d'orienter ses recherches. La plupart des évaluations avaient donc une partie contenant des conseils aux auteurs. Les communications en anglais ont été acceptées, et les évaluations ont été rédigées dans la même langue que l'article.

7 articles longs et 5 articles courts ont été soumis. Deux articles courts supplémentaires ont été soumis à la session jointe des posters et démonstrations de PFIA. Il faut noter que ces deux derniers articles n'ont pas reçu autant d'attention de la part du comité de programme de RJCIA. Nous encourageons donc les jeunes chercheurs à soumettre leur article court à RJCIA plutôt qu'à la session posters et démonstrations de PFIA. Les contraintes concernant le nombre de page limite ont été supprimées pour la version finale de l'article, afin de permettre aux auteurs d'ajouter du contenu ou des références bibliographiques lorsque c'était suggéré par les évaluateurs. Un article long et un article court n'ont pas été inclus dans les actes ni le programme, car aucun de leurs auteurs n'a pu s'inscrire à la Plate-Forme. Le programme de RJCIA contient finalement 6 articles longs, et 6 articles courts. Les articles courts ont été présentés sur posters, et à l'oral lors d'une session *3-minutes madness*.

Un effort supplémentaire serait à fournir de la part des membres du comité de programme pour encourager les jeunes chercheurs de leurs communautés respectives à soumettre à RJCIA.

Nous avons eu le plaisir d'accueillir [Cassia Trojahn, Maître de Conférences du l'IRIT, Toulouse](#), pour une présentation invitée contenant de nombreux conseils à l'attention des jeunes chercheurs intitulée : *How to write a good state of the art : should it be the first step of your thesis ?*

Je remercie l'Association Française pour l'Intelligence Artificielle (AFIA) qui porte l'organisation de PFIA et des conférences hébergées, ainsi que les organisateurs locaux au l'IRIT (laboratoire fondateur de l'Institut 3IA toulousain ANITI) pour la gestion du site web, l'édition du template des actes, et tous les aspects logistiques. Je remercie Florent Breuil pour son aide à la création des actes sur HAL. Je remercie l'ensemble des membres du comité de programme, dont les noms figurent ci-après. Finalement, je remercie les jeunes auteurs sans qui cette conférence ne serait pas possible.

Maxime Lefrançois

Comité de programme

Président

- Maxime Lefrançois (MINES Saint-Étienne)

Membres

- Céline Alec (Université Caen-Normandie)
- Jonathan Bonnet (Continental)
- Mathieu Chollet (University of Glasgow)
- Nicolas Cointe (TU Delft)
- Antoine Cully (Imperial College London)
- Jérôme Delobelle (INRIA Sophia-Antipolis - Méditerranée)
- Dennis Diefenbach (University Jean Monet)
- Géraud Fokou Pelap (INRIA / EDUCLEVER)
- Hugo Gilbert (Gran Sasso Science Institute)
- Pierre-François Gimenez (LAAS)
- Maxime Gueriau (Trinity College Dublin)
- Robin Lamarche-Perrin (CNRS / ISC-PIF / LIP6)
- Marc Law (University of Toronto)
- Amélie Levray (University of Edinburgh)
- Juan Antonio Lossio-Ventura (University of Florida)
- Jean-Guy Mailly (LIPADE, Université Paris Descartes, France)
- Wafa Mefteh (Laboratoire MIRACL, Université de Gabès, Tunisie)
- Amro Najjar (Umeå University, Sweden)
- Abdelkader Ouali (INRIA Nancy - Grand Est)
- Michaël Perrot (Max Planck Institute for Intelligent Systems)
- Eric Piette (DKE)
- Abdallah Saffidine (The University of New South Wales)
- Michael Sioutis (Aalto University)
- Nicolas Szczepanski (CRIL)
- Michaël Thomazo (Inria, DIENS, ENS, CNRS, PSL University)
- Thibaut Vallée (GREYC)
- Nicolas Verstaevel (SMART Infrastructure Facility, University of Wollongong)

I - Présentation invitée

How to write a good state of the art : should it be the first step of your thesis?

How to write a good state of the art: should it be the first step of your thesis ?

Cassia Trojahn

Université de Toulouse, CNRS, Toulouse INP, UT3, UT2

Cassia.Trojahn@irit.fr

Présentation invitée

In scientific writing, the state of the art describes the current knowledge about the studied matter through the analysis of similar or related published work. It might provide a comprehensive overview of what has been done in the field and what should be further investigated, in order to help formulating the problems and hypothesis the thesis intends to address. Producing a good state of the art might be considered the main initial step of a PhD thesis. This is however a challenging task that involves analysing, comparing, evaluating and linking different sources (i.e., many hours of reading and content organisation). This task may be also considered an intimidating task that requires the help and guiding of the thesis supervisors. While starting a thesis by producing (good) state of the art is a nice starting point, this process is non linear and involves many iterations. It requires as well to be continuously and incrementally refreshed. The different types of state of the art (to be incorporated into a paper, a survey article or thesis), the steps of writing a (good) state of the art and some tips facilitating the process will be discussed.

Retrouvez en [cliquant ici](#) ses diapos de présentation.

II - Articles longs

Intention et logique épistémique dynamique

A. Goudyme¹ N. Chetcuti-Sperandio¹ S. Lagrue² T. de Lima¹

¹CRIL UMR CNRS 8188/Université d'Artois

²HEUDIASYC UMR CNRS 7253/Université de Technologie de Compiègne
 {goudyme,chetcuti,delima}@cril.fr sylvain.lagrue@hds.utc.fr

Résumé

Modéliser l'intention permet d'expliquer les décisions prises par des agents. Dans ce travail, nous proposons une modélisation de l'intention dans des jeux épistémiques représentés en logique épistémique dynamique. Étant données une propriété et une suite d'actions déjà réalisée par un joueur dans un tel jeu, nous proposons une méthode capable de déterminer si le joueur a eu l'intention d'obtenir cette propriété. Un exemple de l'utilisation de cette méthode est donné en utilisant une version simplifiée du jeu collaboratif Hanabi.

Mots Clef

Logique épistémique dynamique, intention, jeux épistémiques.

Abstract

Modeling intention is essential to explain decisions made by agents. In this work, we propose a model of intention in epistemic games, represented in dynamic epistemic logic. Given a property and a sequence of actions already performed by a player in such a game, we propose a method able to determine whether the player had the intention to obtain the property. An example of the method is given using a simplified version of the collaborative game Hanabi.

Keywords

Dynamic epistemic logic, intention, epistemic games.

1 Introduction

Pouvoir déterminer le but poursuivi par un agent ou par un groupe d'agents à partir de ses connaissances aurait un intérêt certain dans des domaines tels que l'économie, les jeux et, bien sûr, l'intelligence artificielle. Notre objectif dans cet article est d'être capables de découvrir l'intention des agents à partir, d'une part, des connaissances de ceux-ci sur les actions à leur disposition et, d'autre part, des actions qu'ils ont finalement réalisées. Nous avons choisi de nous restreindre aux jeux épistémiques. L'univers des jeux permet de travailler dans un cadre défini et d'avoir un contrôle total sur les connaissances des joueurs et des actions

qu'ils peuvent effectuer. Les jeux épistémiques désignent des jeux à information incomplète où la réussite au jeu dépend essentiellement des connaissances des joueurs sur l'état du jeu et sur les connaissances des autres joueurs. Des exemples de ce type de jeux sont le Cluedo ou encore Hanabi.

La logique épistémique dynamique est une logique qui traite de la connaissance d'agents et de son évolution à la suite d'événements [1, 2, 4]. Elle apparaît ainsi comme un formalisme adapté à la modélisation logique de l'intention dans les jeux épistémiques, en l'occurrence dans Hanabi, jeu sur lequel s'est porté notre choix pour évaluer nos travaux.

Dans la suite de cet article, nous présenterons tout d'abord la logique épistémique dynamique, puis le jeu Hanabi, avant de détailler nos travaux de modélisation de l'intention dans ce jeu.

2 Logique épistémique dynamique

2.1 Logique modale

La logique modale [3] s'appuie sur les connecteurs de la logique propositionnelle ($\top, \perp, \neg, \wedge, \vee, \rightarrow, \leftrightarrow$) auxquels sont ajoutés deux nouveaux connecteurs représentant la *possibilité* \diamond et la *nécessité* \square . Par exemple, la formule $\diamond\neg(\phi \leftrightarrow \psi)$ signifie « il est possible que ϕ et ψ ne soient pas équivalents ». L'opérateur \diamond peut s'écrire $\neg\square\neg$.

Une des sémantiques les plus utilisées pour la logique modale est celle proposée par Kripke [8]. Un modèle de Kripke se base sur la notion de *mondes possibles*. On note chaque monde M_n (où n est un entier), M représente l'ensemble des mondes M_n . Un *cadre de Kripke* est défini comme l'ensemble formé de M et de R , où R est une relation binaire sur M . Cette relation est appelée *relation d'accessibilité* et définit les mondes accessibles depuis chacun des mondes. Par exemple, dans le système que nous allons utiliser, la relation lie les mondes envisageables pour un joueur. Un cadre de Kripke est souvent représenté sous forme de graphe orienté où les sommets sont les mondes et où les arêtes relient les mondes accessibles.

Enfin, soit \mathbb{P} l'ensemble de propositions, un *modèle de Kripke* est un triplet formé de M , de R et de h , où h est la fonction de valuation des propositions. La fonction de valuation associée à chaque proposition de \mathbb{P} un sous-ensemble de mondes. Un monde peut alors être vu comme un modèle au sens de la logique propositionnelle. Le triplet $U = (M, R, h)$ est aussi appelé *Univers*.

La sémantique de la logique modale est régie par les dix règles suivantes :

1. $\models_{M_j}^U \top$
2. $\not\models_{M_j}^U \perp$
3. $\models_{M_j}^U p$ ssi $M_j \in h(p)$
4. $\models_{M_j}^U \neg\psi$ ssi $\not\models_{M_j}^U \psi$
5. $\models_{M_j}^U \phi \wedge \psi$ ssi $\models_{M_j}^U \phi$ et $\models_{M_j}^U \psi$
6. $\models_{M_j}^U \psi \vee \phi$ ssi $\models_{M_j}^U \psi$ ou $\models_{M_j}^U \phi$ ou les deux
7. $\models_{M_j}^U \psi \rightarrow \phi$ ssi $\models_{M_j}^U \psi$ implique $\models_{M_j}^U \phi$
8. $\models_{M_j}^U \psi \leftrightarrow \phi$ ssi, $\models_{M_j}^U \psi$ ssi $\models_{M_j}^U \phi$
9. $\models_{M_j}^U \Box\psi$ ssi pour tous les mondes M_i de U tels que $M_j R M_i$ $\models_{M_i}^U \psi$
10. $\models_{M_j}^U \Diamond\psi$ ssi pour au moins un des mondes M_i de U tels que $M_j R M_i$ $\models_{M_i}^U \psi$

La règle 3 stipule qu'une variable propositionnelle p est vraie dans un monde si et seulement si ce monde appartient à $h(p)$. Les règles 9 et 10 concernent les modalités : $\Box\phi$ est vraie dans un monde si et seulement si tous les mondes accessibles vérifient ϕ et $\Diamond\phi$ est vraie si au moins un monde accessible vérifie ϕ .

Une formule est *valide* dans un univers U si et seulement si elle est vraie dans tous les mondes de U . Une formule est valide si et seulement si elle est valide dans tous les univers.

Du fait de la sémantique qui vient d'être définie, cette logique modale est dite *normale* puisqu'elle admet les axiomes et règles d'inférence suivants :

1. les tautologies de la logique propositionnelle
2. Kripke (K) : $\Box(\psi \rightarrow \phi) \rightarrow (\Box\psi \rightarrow \Box\phi)$
3. axiome Df : $\neg\Diamond\neg\psi \leftrightarrow \Box\psi$
4. Modus Ponens (MP) : $\frac{\psi \rightarrow \phi, \psi}{\phi}$
5. Nécessitation (RN) : $\frac{\psi}{\Box\psi}$

La règle RN signifie que si ϕ est valide dans un univers, alors $\Box\phi$ est aussi valide dans l'univers (et par extension si ϕ est valide alors $\Box\phi$ est valide).

2.2 Logique épistémique

La logique épistémique est un type de logique modale qui traite de la notion de connaissance et de croyance d'agents [6]. L'agent croit certaines choses et n'a pas d'information sur d'autres. Il imagine alors plusieurs mondes possibles. Appelons N l'ensemble des agents.

Pour chaque agent α , nous introduisons un opérateur (ou connecteur) de croyance/connaissance que l'on peut noter B_α . Pour se situer par rapport à la sous-section précédente, l'opérateur de croyance correspond à l'opérateur \Box . En effet, ce qu'un agent croit doit être vrai dans tous les mondes qu'il imagine. L'opérateur \Diamond est souvent représenté \hat{B}_α . Par ailleurs, chaque opérateur se voit attribuer une relation binaire R_α . La relation R_α lie alors des mondes *indistinguishables* pour l'agent α , c'est-à-dire des mondes dont les propositions communes sont crues par l'agent α . La sémantique de la logique modale ne change pas pour la logique épistémique sauf pour les règles 9 et 10 qui deviennent simplement pour chaque agent α :

- $\models_{M_j}^U B_\alpha\psi$ ssi pour tous les mondes M_i de U tels que $M_j R_\alpha M_i$ $\models_{M_i}^U \psi$
- $\models_{M_j}^U \hat{B}_\alpha\psi$ ssi pour au moins un monde M_i de U tel que $M_j R_\alpha M_i$ $\models_{M_i}^U \psi$

Enfin, les agents peuvent raisonner sur les connaissances des autres agents. En d'autres termes, les agents ont des croyances sur les croyances des autres agents. Et pour aller plus loin, ils peuvent imaginer des mondes qu'ils savent faux mais potentiellement vrais pour d'autres agents. Par exemple, la formule $B_1 B_2 p \wedge \neg B_2 B_1 B_2 p$ signifie que l'agent 1 croit que l'agent 2 croit p mais que l'agent 2 ne croit pas que l'agent 1 croit que l'agent 2 croit p .

2.3 Système S5

La logique épistémique présente deux familles d'axiomes fréquemment utilisées, le système KD45 et le système S5. Le système KD45 traite de croyances, c'est-à-dire, de connaissances qui pourraient être fausses. Le système S5 traite de connaissances vraies et c'est celui que nous allons utiliser.

En effet, dans les jeux, si l'on respecte les règles, les joueurs ne peuvent croire que des vérités, même pour les jeux où le mensonge est autorisé. Dans ce dernier cas, les joueurs, sachant que les autres peuvent mentir, ne prennent pas pour vérité les affirmations des autres joueurs. Le seul moyen pour qu'une connaissance soit fautive est que les règles n'aient pas été respectées.

Le système S5 est un système normal augmenté de deux axiomes :

- axiome 5 : $\neg B_\alpha\psi \rightarrow B_\alpha(\neg B_\alpha\psi)$
- axiome T : $B_\alpha\psi \rightarrow \psi$

L'axiome 5 stipule que l'agent sait ce qu'il ne sait pas. L'axiome T implique que si un agent sait quelque chose, cette chose est forcément vraie. L'ensemble des axiomes du système S5 implique alors que les mondes reliés entre eux par une même relation forment des cliques, que les mondes se relient eux-mêmes et que toutes les relations sont dans les deux sens.

2.4 Actions

On peut rendre la logique épistémique dynamique en y ajoutant des actions. Celles-ci permettent de décrire des changements d'état du monde ou bien l'apprentissage de nouvelles connaissances, pour plus de détails, se référer à [11, 7, 1, 2, 4]. Par exemple, les actions permettent de représenter la pioche d'une carte ou bien l'ouverture d'une porte. L'action transforme un univers en un autre univers. Elle se compose d'une pré-condition et d'une post-condition et fonctionne de la manière suivante : chaque monde du modèle vérifiant la pré-condition est modifié par la post-condition, les autres mondes sont supprimés et les liens entre les mondes restants sont gardés.

Définition 1 Une action est un couple formé d'une pré-condition et d'une post-condition, noté $a = (pre(a), post(a))$. La pré-condition est définie comme une formule du langage. La post-condition est une fonction qui, à chaque proposition $p \in \mathbb{P}$, associe un élément de $\{\top, \perp, p, \neg p\}$.

En particulier, l'action "nop" est définie comme le couple $nop = (pre(nop), post(nop))$, où $pre(nop) = \top$ et $post(nop)(p) = p$ pour tout $p \in \mathbb{P}$. Cette action représente l'action de « ne rien faire », qui n'a pas de pré-condition (i.e., peut être exécutée à tout moment) et ne change pas l'état du monde (i.e., la valeur de chaque proposition p reste la même).

On note A l'ensemble des actions possibles et $U = (M, R, h)$ le modèle épistémique courant. L'exécution de l'action $a \in A$, dans U engendre le modèle $U|_a = (M|_a, R|_a, h|_a)$ où :

- $M|_a = \{M_i \in M \text{ tel que } \models_{M_i}^U pre(a)\}$ est la restriction à l'ensemble des mondes vérifiant la pré-condition de a .
- $R|_a = R \cap (M|_a \times M|_a)$ est la restriction des relations aux mondes de $M|_a$.
- $h|_a(p) = \{M_i \in M \text{ tel que } \models_{M_i}^U post(a)(p)\} \cap M|_a$ est la restriction de la valuation aux mondes de $M|_a$ et réaffectation de la valeur des variables propositionnelles.

Exemple. Soit l'exemple présenté dans la figure 1, l'état épistémique initial U est présenté à gauche. Il y a trois mondes, et une variable propositionnelle p . Le monde M_0 présente un double cerclage : il s'agit du monde réel. Le monde réel est celui qui représente la réalité (c'est-à-dire ce qui est vrai à l'instant actuel). Les mondes M_0 et M_2 appartiennent à $h(p)$, le monde M_1 n'appartient pas à $h(p)$. Soit l'action a définie par la pré-condition $pre(a) = p$ et la post-condition $post(a)(p) = \perp$, l'exécution de l'action a sur l'univers U est présentée à droite : les mondes M_0 et M_2 vérifient la pré-condition, ils sont donc conservés et la variable propositionnelle acquiert une nouvelle valeur, celle de la post-condition.

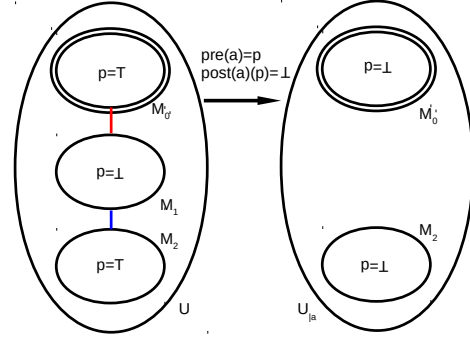


FIGURE 1 – Exemple d'action

3 Hanabi

Hanabi est un jeu de cartes coopératif au tour par tour de 2 à 5 joueurs dont le but est de gagner le plus de points (voir figure 2). Les cartes sont de 5 couleurs différentes (rouge, jaune, bleu, vert, blanc) et de 5 valeurs différentes (de 1 à 5). Pour chaque couleur, il y a trois cartes 1, deux 2, deux 3, deux 4 et une 5. Pour gagner des points, il faut reconstituer les 5 piles de cartes de couleur identique, dans l'ordre croissant des valeurs. On marque 1 point par carte posée, soit un maximum de 25 points. Les joueurs ont en main 4 ou 5 cartes (suivant le nombre de joueurs) ; la particularité de Hanabi est que les joueurs ne voient pas leurs propres cartes mais peuvent voir celles des autres. Pendant son tour, un joueur peut effectuer l'une des trois actions suivantes : donner un indice, défausser une carte, poser une carte. Il existe une pioche, une table, une défausse, ainsi que 8 jetons indices et 3 jetons de vie. Chaque joueur ne pouvant utiliser que ses propres cartes, la difficulté réside donc dans la capacité à donner de bons indices et à poser les bonnes cartes.



FIGURE 2 – Présentation jeu Hanabi

Donner un indice. Un joueur ne peut donner un indice qu'à un seul joueur. Cet indice porte sur l'en-

semble de ses cartes, doit être complet et ne peut concerner qu’une seule des 10 caractéristiques que peuvent avoir les cartes (couleur ou valeur). Par exemple, si le joueur 1 dit au joueur 3 “tes 1ère et 4ème cartes sont bleues” alors les autres cartes ne le sont pas. Enfin, lorsqu’un joueur donne un indice, un jeton indice est consommé. Cette action n’est donc possible que s’il reste des jetons indices.

Défausser une carte. Le joueur prend une carte de sa main et la pose sur la table, face visible. Cette action permet de piocher une nouvelle carte, s’il en reste, et de regagner un jeton indice.

Poser une carte. Lorsque le joueur choisit de poser l’une de ses cartes, il existe deux cas de figure, soit la carte permet d’alimenter l’une des piles, soit elle ne le permet pas et finit dans la défausse, face visible, entraînant la perte d’un jeton de vie. Une carte peut être posée sur la pile de sa couleur si la carte au sommet de cette pile est la carte précédente (de valeur immédiatement inférieure). Une carte de valeur 1 peut être posée si elle permet de débiter une pile d’une nouvelle couleur. Enfin, poser une carte de valeur 5 permet de compléter une pile et de gagner un jeton de vie.

Fin de la partie. La partie se termine s’il ne reste plus de jeton de vie, si le nombre maximal de points est atteint ou après un tour complet suivant la pioche de la dernière carte de la pioche.

4 Intention

4.1 Principes

L’objectif de notre travail est de déterminer *a posteriori* les intentions qu’un joueur a eues au cours d’une partie. En d’autres termes, nous souhaitons expliquer les actions effectuées par un joueur lors d’une partie. Notre idée se base sur le principe suivant : *par rapport à tous les résultats potentiels imaginés par le joueur, a-t-il effectué l’action qui le menait vers le meilleur résultat espéré* ? Dans la suite, nous allons expliquer cette idée plus en détail.

Soit un joueur et une propriété p , exprimée comme une formule en logique propositionnelle.

Premier principe. Nous associons à un univers U une valeur $v(U, p)$ appelée fréquence de la propriété p qui, comme son nom l’indique, est égale à la fréquence des mondes vérifiant p , indistinguables du monde réel pour le joueur (voir figure 3). Cette idée a été initialement proposée par Markus Eger dans sa thèse [5].

Deuxième principe. Supposons que le joueur n’imagine qu’un monde possible, le monde réel, et qu’il a deux actions possibles a et b (voir figure 4). L’action a mène vers un univers de valeur 0 et l’action b vers un univers de valeur 1. La meilleure action pour la propriété p est donc l’action b . Si le joueur a effectué l’action b , alors il avait l’intention d’obtenir la propriété p ,

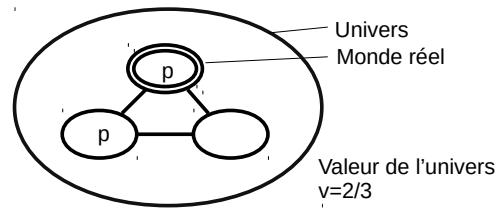


FIGURE 3 – Illustration du premier principe

dans le cas contraire, il n’en avait pas l’intention. En outre, on affecte à l’univers initial la valeur de l’univers obtenu par la meilleure action.

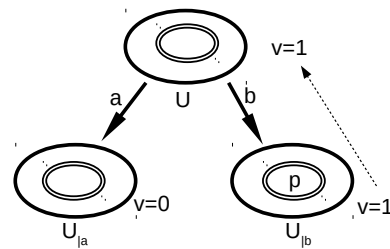


FIGURE 4 – Illustration du deuxième principe simple

Supposons maintenant que le joueur imagine deux mondes possibles, avec toujours deux actions a et b (voir figure 5). Dans le premier monde, l’action a mène vers un univers de valeur 0,5 et l’action b vers un univers de valeur 1. Dans le second monde, l’action a mène vers un univers de valeur 0,5 et l’action b vers un univers de valeur 0. Les mondes étant équivalents pour le joueur, il y a autant de chance pour lui de se trouver dans l’un que dans l’autre. On associe donc aux actions la moyenne des valeurs des univers vers lesquels elles mènent. La valeur de a est donc de 0,5 et la valeur de b est de 0,5 également. Comme auparavant, nous affectons à l’univers initial la valeur de la meilleure action (ou des meilleures actions comme ici). Dans le cas présent, peu importe quelle action a été exécutée, l’obtention de la propriété p ne peut pas être considérée comme intentionnelle puisque toutes les actions permettent de l’obtenir avec autant de chance.

En revanche, l’intention d’obtenir une propriété est claire dans le cas où il n’y a qu’une seule meilleure action pour cela. Pour d’autres pourcentages de meilleures actions parmi l’ensemble des actions, il est

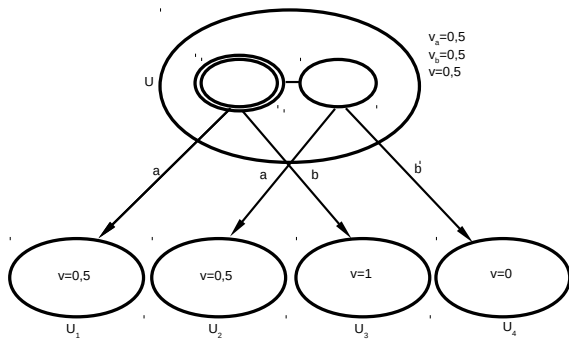


FIGURE 5 – Illustration du deuxième principe

plus difficile de mesurer l'intention . Ainsi s'il semble toujours pertinent de parler d'intention lorsqu'il y a 2 meilleures actions parmi 10 actions pour obtenir une propriété, pour 5 meilleures actions parmi 10 on pourrait parler d'aléatoire. Et à 9 actions parmi 10, il est difficile de parler d'intention d'obtenir la propriété. Cependant, dans ce dernier cas, nous avons alors une meilleure action parmi 10 pour obtenir la non-propriété. Établir des seuils d'intention fait partie de nos réflexions actuelles.

Troisième principe. Dans un troisième exemple, le joueur imagine de nouveau deux mondes, mais cette fois-ci, c'est à un autre joueur de jouer et celui-ci a trois actions possibles : a , b et c (voir figure 6). Le second joueur ayant ses propres objectifs, les différentes actions sont plus ou moins intéressantes au regard de ses objectifs personnels et on appelle *utilité* la mesure de cet intérêt. Cependant ici, ce qui sera pris en compte n'est pas l'utilité définie par le second joueur mais ce que le premier joueur imagine être l'utilité pour le second joueur.

Supposons d'abord que le premier monde est le monde réel, le premier joueur imagine l'utilité des actions, par exemple $u(a) = 0,6$, $u(b) = 0,6$ et $u(c) = 0,3$. Le premier joueur suppose que le second joueur est rationnel et qu'il va donc effectuer une action d'utilité maximale. Les deux actions a et b ayant une utilité maximale, le premier joueur pense que le second joueur effectuera l'une de ces deux actions avec une équiprobabilité. Ces deux actions mènent vers deux univers différents dont les valeurs sont de 0,8 et 0,6. La moyenne de la valeur de ces deux actions, qui est 0,7, est affectée au premier monde. Nous effectuons à présent le même raisonnement avec le second monde. Les actions ont des utilités différentes, les univers ainsi générés ont des valeurs différentes et, finalement, nous trouvons, pour ce second monde, une valeur de 0,3. Pour terminer, nous faisons la moyenne des valeurs

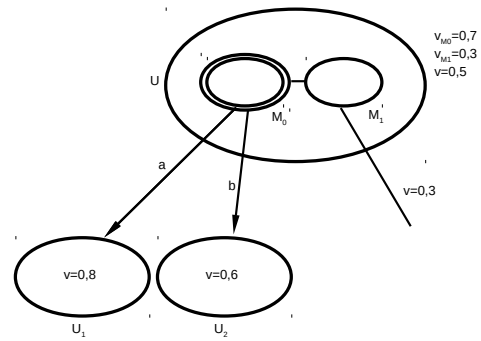


FIGURE 6 – Illustration du troisième principe

des deux mondes, ce qui donne 0,5 et nous affectons cette valeur à l'univers.

La méthode de détermination de l'intention d'un joueur que nous proposons s'appuiera sur ces 3 principes.

Exemple. Soient 3 joueurs et une propriété p arbitraire. On souhaite déterminer la meilleure action pour le joueur 1 pour obtenir la propriété p en 3 tours. On commence par développer l'arbre des univers comme décrit précédemment mais sur 3 tours ; cet arbre comporte 4 niveaux. Le 1^{er} niveau est composé de l'univers initial. Le 2^{me} niveau est composé de l'ensemble des univers imaginés par le joueur 1 après son tour. Le 3^{me} niveau est composé de l'ensemble des univers possibles pour le joueur 1 après son tour et celui du joueur 2. Le 4^{me} niveau est composé de l'ensemble des univers possibles pour le joueur 1 après le tour des joueurs 1, 2 et 3. On calcule la valeur des univers du 4^{me} niveau à l'aide du premier principe. Ceci nous permet de calculer ensuite la valeur des univers du 3^{me} et du 2^{me} niveaux grâce au troisième principe. Enfin, la valeur de l'univers du 1^{er} niveau est calculée au moyen du deuxième principe. Ceci permet de vérifier si l'action effectuée par le joueur 1 est celle qui, du point de vue du joueur en question, avait une meilleure chance d'obtenir la propriété p .

Remarque. La notion de valeur d'un univers est proche de celle d'utilité. En effet, pour les exemples les plus simples, la valeur de l'univers est simplement la fréquence d'apparition de la propriété recherchée au sein de l'univers, en d'autres termes, la probabilité de l'obtenir. Étant donné qu'un univers est lié à l'action qui permet de l'atteindre, on peut attribuer à chaque action une mesure de son utilité pour obtenir la propriété. Cela définit donc une utilité. Cependant, nous préférons ne pas utiliser ce terme car nous donnons des valeurs à des états et non pas à des actions.

4.2 L'intention dans les jeux épistémiques

La notion d'agent devient ici plus importante, puisqu'elle n'est plus seulement caractéristique des relations d'accessibilité entre les mondes mais également des actions possibles. En effet, dans la présentation de la logique épistémique dynamique, la notion d'agent apparaît uniquement pour donner une signification aux différentes relations d'accessibilité (par exemple, la relation d'accessibilité R_3 représente les connaissances de l'agent 3). Il n'est pas utile de préciser la relation qui se rapporte à un agent en particulier, les raisonnements sur la relation suffisent. Ce que nous allons faire ici est d'associer à la fois un agent à sa relation et aux actions qu'il va entreprendre. Il est donc important de construire un système possédant un lien entre l'ensemble des agents, l'ensemble des relations et l'ensemble des actions. On s'attarde donc à séparer les entités, il faut donc ré-associer les agents et leurs relations d'accessibilité, c'est ce qui est décrit dans les deux définitions suivantes :

Définition 2 Soit M un ensemble de mondes, N un ensemble d'agents et R^M un ensemble de relations d'accessibilité sur M vérifiant $\text{Card}(R^M) = \text{Card}(N)$. On appelle fonction d'association des relations entre N et R^M toute fonction bijective de N vers R^M .

La définition 2 permet d'associer une relation d'accessibilité à tout agent et inversement.

Définition 3 Soit \mathbb{P} un ensemble de propositions, M un ensemble de mondes, N un ensemble d'agents, h une fonction de valuation, R^M un ensemble de relations d'accessibilité sur M tel que $\text{Card}(R^M) = \text{Card}(N)$ et f_r une fonction d'association des relations entre N et R^M . On appelle univers le 6-uplet $U = (M, N, R^M, f_r, \mathbb{P}, h)$.

Pour des raisons de lisibilité, on notera indifféremment $f_r(\alpha)$ et R_α la relation d'accessibilité liée à l'agent α . Pour rappel, dans ce système, la relation R_α lie les mondes indistinguables pour le joueur α , pour qui ils ont donc tous autant de chances d'être le monde réel. Attribuer le concept de tour de jeu à la logique épistémique peut se révéler compliqué et contraignant pour les définitions. Introduire les actions simultanées de tous les joueurs (*action jointe*) a été préféré. La notion de tour de jeu est alors modélisée en utilisant l'action *nop* : à chaque tour, un joueur joue pendant que les autres effectuent l'action *nop* (ne font rien). La définition suivante d'action jointe se base sur la logique ATDEL [10].

Définition 4 Soit \mathbb{P} un ensemble de propositions et A un ensemble d'actions sur \mathbb{P} . On note A^n l'ensemble formé par $A^n = \times_1^n A$. On appelle $a^n = (a_1, a_2, \dots, a_n)$

action jointe un élément de A^n défini par une pré-condition jointe et une post-condition jointe noté $a^n = (\text{pre}(a^n), \text{post}(a^n))$. La pré-condition jointe vérifie $\text{pre}(a^n) = \bigwedge_{i=1}^n \text{pre}(a_i)$. La post-condition jointe vérifie, pour chaque élément p de \mathbb{P} :

$$\text{post}(a^n)(p) = \begin{cases} \top & \text{si } \forall a_i, \text{post}(a_i)(p) = \top \\ \perp & \text{si } \forall a_i, \text{post}(a_i)(p) = \perp \\ \neg p & \text{si } \forall a_i, \text{post}(a_i)(p) = \neg p \\ \text{sinon } p & \end{cases}$$

Pour simplifier, nous emploierons a à la place de a^n lorsqu'il n'y aura pas d'ambiguïté.

Définition 5 Soit un univers $U = (M, N, R^M, f_r, \mathbb{P}, h)$, un ensemble A d'actions et une action jointe $a \in A^n$ où $n = \text{card}(N)$. L'univers $U|_a = (M|_a, N, R^M|_a, f_r, \mathbb{P}, h|_a)$ engendré par l'exécution de l'action jointe a vérifie :

- $M|_a = \{M_i \in M \text{ tel que } \models_{M_i}^U \text{pre}(a)\}$ est la restriction à l'ensemble des mondes vérifiant la pré-condition de a .
- $\forall \alpha \in N$ on a $R_{\alpha|_a} = R_\alpha \cap (M|_a \times M|_a)$ est la restriction des relations aux mondes de $M|_a$
- $h|_a(p) = \{M_i \in M \text{ tel que } \models_{M_i}^U \text{post}(a)(p)\} \cap M|_a$ est la restriction de la valuation aux mondes de $M|_a$ et réaffectation de la valeur des variables propositionnelles

Vient ensuite le concept de l'utilité des actions des autres joueurs vu par le joueur qui nous intéresse. On définit donc une fonction d'utilité qui va associer à chaque monde et à chaque joueur une valeur.

Définition 6 Soit A un ensemble fini d'actions et $U = (M, N, R^M, f_r, \mathbb{P}, h)$ un univers. On appelle fonction d'utilité u toute fonction qui associe à un triplet $(a, \alpha, M_i) \in A \times N \times M$ un réel $x \in \mathbb{R}$. On appelle Ω Univers Actionnable Mesuré (UAM) tout triplet (U, A, u) .

Définition 7 Soit $\Omega = (U, A, u)$, un monde $M_i \in M$ et un agent $\alpha \in N$. On note $A_{M_i, \alpha}^\Omega$ l'ensemble des actions les plus utiles pour α pour le monde M_i où $A_{M_i, \alpha}^\Omega = \{a \in A \mid \forall b \in A, u(a, \alpha, M_i) \geq u(b, \alpha, M_i)\}$

Proposition 1 Soit $\Omega = (U, A, u)$ et $\alpha \in N$. $\forall M_i \in M$, l'ensemble $A_{M_i, \alpha}^\Omega$ est non vide et fini.

Preuve : Nous avons $A_{M_i, \alpha}^\Omega \subseteq A$. Comme A est fini, alors $A_{M_i, \alpha}^\Omega$ est fini. De plus, comme A est fini, $\forall \alpha \in N, \forall M_i \in M, \{u(a, \alpha, M_i) \mid a \in A\}$ est fini et possède donc un plus grand élément. De ce fait, $A_{M_i, \alpha}^\Omega$ est non vide. QED.

À présent, nous allons définir formellement le premier principe, c'est-à-dire le calcul de la fréquence d'une variable propositionnelle au sein des mondes indistinguables du monde réel pour le joueur.

Définition 8 Soit $U = (M, N, R^M, fr, \mathbb{P}, h)$ un univers, $M_0 \in M$ et $\alpha \in N$. On note U^{M_0} l'univers tel que M_0 est le monde réel et on appelle $I_\alpha^{U^{M_0}}$ ensemble des mondes envisageables de 1^{er} degré pour α l'ensemble de tous les mondes $M_i \in M$ tel que $M_0 R_\alpha M_i$.

Définition 9 Soit un univers U^{M_0} , une formule p et un agent α . On appelle $pr_\alpha^{U^{M_0}}(p)$ rapport de présence de p pour α le rapport : $pr_\alpha^{U^{M_0}}(p) = \frac{Card(I_\alpha^{U^{M_0}} \cap h(p))}{Card(I_\alpha^{U^{M_0}})}$ si $I_\alpha^{U^{M_0}} \neq \emptyset$ sinon $pr_\alpha^{U^{M_0}}(p) = 0$

Ce qui nous permet de calculer la valeur de l'univers suivant une propriété pour k tours de jeu à l'avance.

Définition 10 Soit un univers actionnable mesuré $\Omega = (U^{M_0}, A, u)$, un ensemble d'actions A , un agent $\alpha \in N$ et un entier positif k . On appelle $v(U^{M_0}, A, u, \alpha, p, k)$ valeur de l'univers U^{M_0} pour l'agent α à l'ordre k le réel suivant, avec $A_{M_i}^\alpha = A_{M_i,1}^\alpha \times A_{M_i,\alpha-1}^\alpha \times A_{M_i,\alpha+1}^\alpha \times A_{M_i,n}^\alpha$: $v(U^{M_0}, A, u, \alpha, p, k \geq 1) = \max_{a \in A} (moy_{monde})$ où $moy_{monde} = \sum_{M_i \in I_\alpha^{U^{M_0}}} moy_{action} \times \frac{1}{\|I_\alpha^{U^{M_0}}\|}$ où $moy_{action} = \sum_{(a_1, \dots, a_{n-1}) \in A_{M_i}^\alpha} valeur \times \frac{1}{\|A_{M_i}^\alpha\|}$ où $valeur = v(U_{|(a_1, \dots, a_n)}^{M_i}, A, u, \alpha, p, k - 1)$ et $v(U^{M_0}, A, u, \alpha, p, 0) = pr_\alpha^{U^{M_0}}(p)$

Le maximum caractérise le deuxième principe, tandis que la double somme caractérise le troisième principe. Lorsque c'est le tour du joueur qui nous intéresse, la seule action exécutée par les autres joueurs est l'action *nop*. La deuxième somme ne possède alors qu'un seul élément, il n'y a donc pas de moyenne sur les actions des autres joueurs, ce qui correspond au deuxième principe. Lorsque c'est le tour de l'un des autres joueurs, la seule action maximum pour le joueur est l'action *nop*, ses autres actions donnant des univers vides et donc des valeurs nulles. Il y a donc une moyenne sur les actions et sur les mondes, ceci correspond au troisième principe.

Définition 11 Soit un univers actionnable mesuré $\Omega = (U^{M_0}, A, u)$, un ensemble d'actions A , un agent $\alpha \in N$, une variable propositionnelle $p \in \mathbb{P}$ et un entier strictement positif k , on appelle $Best(\Omega, p, \alpha, k)$ ensemble des meilleures actions pour obtenir une propriété p au k -ième tour pour le joueur α l'ensemble suivant : $Best(\Omega, p, \alpha, k) = \{a \in A \mid \forall b \in A, v_a(U^{M_0}, A, u, \alpha, p, k) \geq v_b(U^{M_0}, A, u, \alpha, p, k)\}$

Ces deux dernières définitions permettent de caractériser les meilleures actions suivant le joueur pour obtenir une propriété k tours de jeu à l'avance. Ainsi, en se donnant une propriété p , nous pouvons vérifier si le joueur a effectué la meilleure suite d'actions pour l'obtenir. Si tel est le cas, alors l'obtention de p est

intentionnelle, sinon nous considérons que p a été obtenue par accident. Par exemple, si l'agent α a effectué l'action a et si $a \in Best(\Omega, p, \alpha, k)$ alors cela était intentionnel. Cela devient faux si toutes les actions font partie des meilleures actions ($Best(\Omega, p, \alpha, k) = A$), dans ce cas aucune action n'est intentionnelle.

4.3 Utilité

Le calcul de la meilleure action demande de définir une utilité pour les actions, ou plus exactement, une utilité du point de vue du joueur pour les actions des autres joueurs. Trouver la meilleure façon de modéliser l'utilité pour un jeu est un travail de recherche en lui-même et ce n'est pas celui que nous avons voulu effectuer ici. Nous proposons donc une définition d'utilité basée sur la récompense de fin de partie qui est calculable par n'importe quel joueur pour n'importe quel autre joueur dès lors que l'information sur la récompense de fin de partie est accessible à tous les joueurs.

Le joueur imagine plusieurs mondes possibles et pour chaque monde qu'il imagine, les autres joueurs peuvent imaginer d'autres mondes. Dans la figure 7, pour le joueur 1, le monde M_0 et le monde M_1 sont indistinguables du monde réel M_0 . Pour le joueur 2, les mondes M_0 et M_5 sont indistinguables et les mondes M_1 et M_2 sont indistinguables.

Nous commençons par définir l'utilité des actions pour le joueur 2 d'après le joueur 1. Le joueur 1 sait que le joueur 2 hésite soit entre la clique de mondes M_0 et M_5 , soit entre la clique de mondes M_1 et M_2 . Il faut donc traiter ces deux cliques séparément. Nous nous focalisons d'abord sur la première clique. Pour calculer l'utilité d'une action du joueur 2 suivant le monde de la clique, on effectue le raisonnement suivant :

- on suppose que ce monde est le monde réel ;
- on regarde le score maximum que l'on pourrait atteindre à la fin du jeu après avoir effectué l'action jointe du joueur 2 et des autres joueurs (ici le joueur 1 seulement) ;
- on calcule la moyenne sur l'ensemble des actions des autres joueurs (ici le joueur 1 seulement) ;
- on définit cette moyenne comme l'utilité de l'action pour ce monde.

Comme les joueurs envisagent ces mondes de manière équiprobable, pour comparer les actions, il faut calculer la moyenne des utilités sur l'ensemble des mondes de la clique. On obtient l'utilité de l'action pour cette clique. Enfin, on définit l'utilité d'une action du joueur 2 d'un monde possible pour le joueur 1 comme étant égale à l'utilité de la clique qu'elle génère pour le joueur 2. Pour les autres mondes, la valeur de l'utilité, n'ayant pas d'importance, est fixée arbitrairement.

Pour ce qui concerne les utilités du joueur 1 pour le joueur 1 elles n'ont pas d'importance, elles sont donc aussi fixées arbitrairement.

Pour les jeux au tour par tour, seule l'action *nop* ne

donne pas un univers vide lorsque ce n'est pas au joueur de jouer. On ne peut donc pas à priori donner de valeurs à l'utilité pour les autres actions. Donner une valeur identique à chaque univers vide ne changerait pas l'ordre des valeurs des utilités des actions. On peut aussi sauter cette étape de calcul de moyenne sur les actions des autres joueurs. Dans les deux cas, l'action de plus grande utilité sera la même.

4.4 Exemple

Nous allons utiliser le jeu Hanabi dans une version simplifiée pour illustrer l'utilisation des définitions précédemment données. Dans le jeu simplifié, il n'y a plus que les cartes 1, 2 et 3 rouges et deux joueurs, il y a toujours 8 jetons indices et 3 jetons de vie. Il n'y a plus que trois actions (D, P et I) : défausser sa carte, poser sa carte, donner un indice (qui correspond ici à donner la valeur de la carte de l'autre). Lors de la donne, le joueur 1 avait reçu la carte 1, le joueur 2 avait reçu la carte 2 et la pioche contenait donc le 3. La figure 7 présente l'univers initial. La partie s'est déroulée, le joueur 1 a commencé par donner un indice "tu as la carte 2" et après le tour du joueur 2, le joueur 1 connaît sa carte et la carte 2 n'est pas défaussée. Nous allons donc vérifier si la propriété « le joueur 2 n'a pas défaussé et le joueur 1 connaît sa carte », noté $p = \neg D : 2 \wedge (B_1 J1 : 1 \vee B_1 J1 : 2 \vee B_1 J1 : 3)$ a été obtenue intentionnellement.

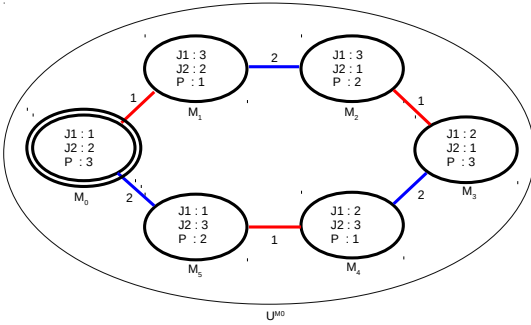


FIGURE 7 – Univers initial

À partir de l'univers initial, figure 7, le joueur 1 imagine deux mondes possibles : M_0 et M_1 . Nous devons donc considérer deux cas de figure : l'un où le monde réel est M_0 et l'autre où le monde réel est M_1 (voir figure 8). Nous calculons la valeur de l'action « donner un indice ». Le joueur 1 va déclarer « tu as la carte de valeur 2 » ceci a pour conséquence que les mondes ne vérifiant pas « le joueur 2 à la carte de valeur 2 » disparaissent. Dans cet exemple, effectuer cette action depuis le monde M_0 (respectivement le monde M_1) donne l'univers $U_{|I}^{M_0}$ (respectivement

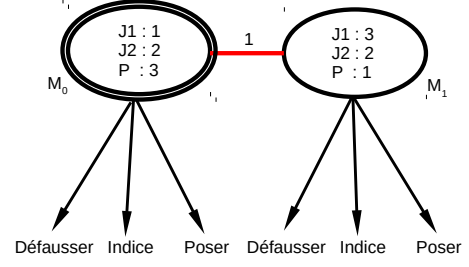


FIGURE 8 – Initialisation du calcul de la valeur des actions du joueur 1

$U_{|I}^{M_1}$, seul le monde dit "réel" diffère) présenté en figure 9. Dans cet univers $U_{|I}^{M_0}$, le joueur 2 a les trois mêmes actions possibles. Si le monde M_0 est le monde réel, l'action D et P ont une utilité de 1 et l'action I a une utilité de 3. Le joueur 1 pense donc que le joueur 2 va effectuer cette action, qui permettra d'obtenir la propriété avec une fréquence de 1. Si le monde M_1 est le monde réel, les trois actions D, P et I ont une utilité de 1. Pour le joueur 1, le joueur 2 va effectuer équiprobablement l'une de ces trois actions. Les actions D et P ne permettent pas d'obtenir la propriété, l'action I permet d'obtenir la propriété avec une fréquence de 1. La valeur de cet univers est donc : $\frac{1+1+3}{2}$.

On effectue le même raisonnement pour $U_{|I}^{M_1}$, qui, dans cet exemple, présente les mêmes calculs que $U_{|I}^{M_0}$. En calculant la moyenne de la valeur de ces deux univers on obtient la valeur de l'action « donner un indice » :

$$v_{Indice}(U^{M_0}, A, u, 1, p, 2) = \frac{\frac{1+1+3}{2} + \frac{1+1+3}{2}}{2} = \frac{2}{3}$$

De même, on trouve $v_{Defausser}(U^{M_0}, A, u, 1, p, 2) = \frac{1}{3}$ et $v_{Poser}(U^{M_0}, A, u, 1, p, 2) = \frac{2}{3}$. Les meilleures actions sont donc l'action de donner un indice et l'action de poser. Le joueur avait donc bien l'intention d'obtenir la propriété « je connais ma carte, et le joueur 2 n'a pas défaussé » au bout de deux tours.

5 Conclusion

Nous avons défini une première modélisation de l'intention puisqu'il nous est à présent possible de vérifier si une propriété donnée a été obtenue intentionnellement ou non.

Ceci n'est pas le premier travail qui intègre intention et logique épistémique. Le travail de Lorini et Herzig [12] modélise l'intention via des opérateurs de tentatives réussies ou échouées. Toutefois, leur logique modélise

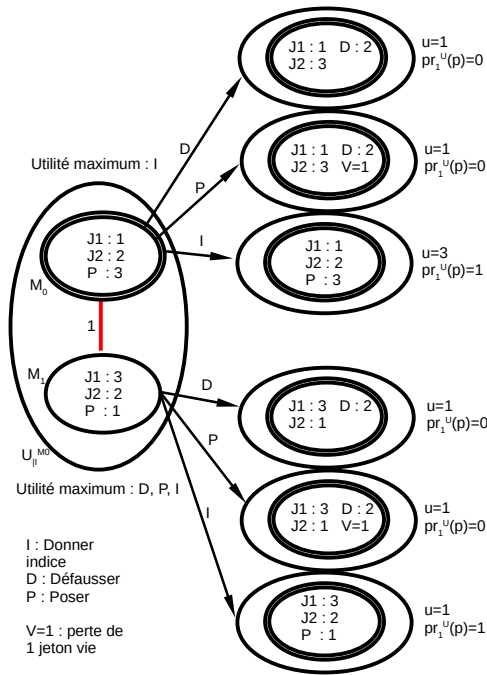


FIGURE 9 – Calcul de la valeur de l'action « donner un indice » pour le monde M_0

le passage du temps avec une ligne droite (c'est-à-dire qu'il n'y a qu'un seul futur possible), il n'y a pas d'embranchements. Il est donc beaucoup moins naturel de capturer une sémantique de jeu.

Dans notre approche, il est à noter que plus on prend une formule spécifique à l'univers dans lequel on est arrivé, plus il est certain que les actions seront considérées comme intentionnelles. Une formule plus générale permettrait donc a priori de mieux caractériser l'intention. La définition de la généralité d'une formule au sein d'un ensemble d'univers est une des voies pour continuer le travail présenté dans cet article.

Par ailleurs, les mondes sont ici considérés comme équiprobables, il pourrait être intéressant, dans des travaux futurs, d'intégrer des logiques pondérées comme celle présentée par Legastelois [13].

Notre modélisation prend en compte des joueurs capables d'imaginer tous les mondes possibles et tous les univers qui en découleraient suivant les actions qu'ils pensent les plus pertinentes. Si une machine a déjà des ressources limitées, ce même travail pour un humain est encore plus difficile. L'une des pistes envisagées pour tenir compte de cette limitation serait de considérer une action comme meilleure à partir du moment où sa valeur dépasse un seuil donné (par exemple, 80% de la véritable valeur maximale).

Un logiciel est en cours de développement, il nous permettra de confronter les concepts que nous avons

construits à la réalité du jeu Hanabi. Il permettra également de tester l'utilité que nous avons définie et de voir s'il est possible de retrouver des stratégies telles que celles définies dans [14].

Références

- [1] A. Baltag and L.S. Moss. *Logic for epistemic programs* Synthese, 139 :165-224, 2004
- [2] A. Baltag and L.S. Moss. and S. Solecki. *The logic of common knowledge, public announcements and private suspicions* Technical Report SEN-R9922, CWI, Amsterdam, 1999
- [3] B.F. Chellas. *Modal Logic*, Cambridge University Press, pp. 3-160, 1980.
- [4] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*, Springer, 2007.
- [5] M. Eger. *Intentional Agents for Doxastic Games*, North Carolina State University, 2018
- [6] R. Fagin and J.Y Halpern and Y. Moses and M.Y. Vardi. *Reasoning about knowledge*, MIT Press, pp. 1-232, 2003.
- [7] J.D. Gerbrandy and W. Groeneveld. *Reasoning about information change*, Journal of Logic, Language, and Information, 6 :147-169, 1997
- [8] S.A. Kripke. *Semantical Analysis of Modal Logic*, The Journal of Symbolic Logic, 1959.
- [9] D.K. Lewis. *Convention : A Philosophical Study*, 1969
- [10] T. de Lima. *Alternating-time temporal dynamic epistemic logic*, Journal of Logic and Computation, vol 24, pp. 1145-1178, 2014
- [11] J.M. Plaza. *Logics of public communications* Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems, pp. 201-2016, 1989
- [12] E. Lorini and A. Herzig *A logic of intention and attempt* Kluwer Academic Publishers, 2007
- [13] B. Legastelois *Extension pondérée des logiques modales dans le cadre des croyances graduelles* Université Pierre et Marie Curie - Paris VI, 2017
- [14] B. Bouzy *Playing Hanabi Near-Optimally* Université Descartes - Paris V, 2017

Towards a multi-agent model to prevent damage caused by cocoa mirids to cocoa pods

E. D. Ngounou Ntougam¹ V. C. Kamla¹ Y. Mualla²
I. H. Tchappi^{1,2} S. Galland² J. C. Kamgang¹
S. Emvudu Wono³

¹ University of Ngaoundere, 455 Ngaoundere, Cameroon

² CIAD, Univ. Bourgogne Franche-Comté, UTBM, F-90010 Belfort, France

³ University of Yaounde, Yaounde, Cameroon

ngounoudimitry@gmail.com

Abstract

Agriculture in general and cocoa farm in particular is one of the primary sources of income for several African countries. Although quite useful for the development of Africa, cocoa production is faced with several challenges like the diseases of cocoa trees. Mirid (Sahlbergella singularis) is the worst pest affecting cocoa production across west Africa causing 30% to 40% of damage on cocoa production. It is difficult to control the mirids' population and its impact on the cacao production. Related works on mirids studied the mirids population as a macroscopic entity without considering local interactions of individual mirids. These models did not consider the behaviours and the environment of mirids which are essential due to their complexity. To overcome these drawbacks, this paper proposes an agent-based model of mirids considering the individual interactions of mirids within the environment. The proposed model consists of agents representing the mirids through their life cycle in cocoa farm environments. It is based on the biological and ecological partial knowledge found in the literature, and is built according to the GAIA methodology. The main assets of this model are the reproduction of the dynamics of the mirids on a cocoa pod and damage caused by them at the micro level for human users (entomologists, agricultural engineers, farmers and decision makers) to thoroughly understand the phenomena and the emergent properties outcomes.

Keywords

Mirid, Agent-based modeling, Cocoa, GAIA methodology.

Résumé

L'agriculture en général et la culture du cacao en particulier est l'une des principales sources de revenus de

plusieurs pays africains. Cependant, la production du cacao est confrontée à plusieurs défis tels que les maladies et l'attaque des insectes nuisibles. Les mirides (Sahlbergella singularis) sont les insectes les plus préjudiciables à la culture du cacao en Afrique. Avec 30 à 40 % de dommages sur la production, il est difficile de contrôler la population des mirides ainsi que son impact sur la production de cacao. Les travaux précédents sur les mirides, étudient la population de cet insecte selon le point de vue macroscopique sans tenir compte des interactions locales entre les mirides. Ces modèles ne prennent pas en compte les comportements individuels des mirides et leurs interactions avec l'environnement qui sont essentiels en raison de leurs complexités. Pour remédier à ces insuffisances, le présent article propose un modèle de mirides à base d'agents prenant en compte les interactions individuelles des mirides dans leur environnement. Le modèle proposé est constitué d'agents représentant le cycle de vie du miride sur une cabosse de cacao. Il est basé sur les connaissances biologiques et écologiques du miride, et est construit selon la méthodologie GAIA. Ces modèles ainsi que leurs implémentations sont destinés à des utilisateurs tels que les entomologistes, les ingénieurs agronomes, les agriculteurs et les décideurs. Cela leur permettra de mieux comprendre les phénomènes émergents dues aux mirides ainsi que leurs propriétés.

Mots Clés

Mirid, Modélisation à base d'agents, Cacao, Méthodologie GAIA.

1 Introduction

With more than 72% of worldwide cocoa production, Africa (Ivory Coast, Ghana, Nigeria and Cameroon) is currently the main producer of cocoa [22]. Cocoa is cultivated by smallholder with lower livelihood, yet it

is an important source of revenue for these farmers.

African production faces numerous difficulties like : ageing of farmers, ageing of plantations, non-respect of good agricultural practices, presence of diseases and pests [11]. Cocoa Swollen Shoot Virus (CSSV) and cocoa Black Pod Rot (BPR) are the two diseases that most seriously affect African cocoa production [7]. Additionally, *Sahlbergella singularis* and *Distantiella theobroma* are the two pests that most negatively impact production of cacao in all African producer countries [6]. Between these two species, *Sahlbergella singularis* is the most common and most harmful [6]. These two species, also called mirid bugs, feed principally on cocoa (*Theobroma cacao*). Mirids feed on pods and shoots and they cause the drying up of leaves and a quick destruction of the cocoa canopy. The immature (five instars) and adult stages of these sucking insects feed on the sap of young branches, by injecting a digestive saliva [3].

Due to the considerable thickness of the pod cortex, damages caused by mirids are relatively superficial on pods and there is no direct link between the intensity of feeding lesions on mature pods and bean quality [6]. Nevertheless, young fruits (less than 10 cm in length and less than 2 months old) also called cherelles, can be seriously damaged by the mirid's feeding lesions. These feeding lesions can increase the abortion of young fruits and thus impact production [7].

The main strategy to fight against mirids is the use of insecticide. Although alternative cocoa pest control methods have been developed to fight against mirids. They include cultural management [17], varietal management [1], as well as semi-chemical management with the use of synthetic sexual pheromone traps [1, 25]. However, their effectiveness remains limited. Cocoa is cultivated in tropical agro-forests which are characterized by complex associations of multi-functional and uneven-aged trees and crops [21]. In such environments, it is not obvious to count mirid individuals (immatures and adults) or evaluate the impact of their actions on cocoa plant. Agent-based approaches are suitable to model complex systems [9, 27?]. They seem to be the most relevant option to model the dynamic behavior of mirids at the micro level in order to understand mirids behavior and thus forecast its evolution and adopt an efficiency control strategy. The aim of this work is therefore to propose an agent-based model for the mirids dynamic. To this end, an initial agent-based model is proposed based on the mirid's life cycle. We make some representations of a mirid agent on a cocoa pod to present how it progresses. In order to be logical and clear, we follow the GAIA methodology [28]. In fact, GAIA methodology is well-suited to design and build systems in complex and open environments at both micro and macro levels [9] such as the motion and evolution of mirids in a cocoa

farm.

This work is ongoing and the paper focuses only on the interactions of mirids in the environment. In the future, other interactions will be considered such as the interaction with human users (entomologists, agricultural engineers, farmers and decision makers). Therefore, we shall make sure that the target people will be able to understand our models and simulation results. To do so, the simulation should explain to the users about the behaviours of all agents and their interactions.

The paper is organized as follows : Section 1.1 highlights a brief overview on theoretical models of mirids. Section 2 presents our agent-based model of the life cycle of mirids based on the GAIA methodology. Section 3 discusses the need for explainability and how to achieve it and section 4 concludes the work and gives some prospective directions.

1.1 Related Work

It is important to understand mirid's life cycle and to have information about its biology and ecology. The life cycle of a mirid is of three phases : *egg*, *nymph* and *adult*.

Mirids develop on the cocoa pod. Females insert eggs in cocoa living tissues, and egg development requires healthy tissues to be completed [5]. The embryonic development of a mirid is between 15 and 20 days in average with minimum 9 days and 21 days maximum [5]. After the embryonic phase, the egg becomes a nymph. On average, the life-cycle duration (eggs to adults) of mirids is 40 days with a minimum of 36 days [4] and a maximum of 50 days [5]. The rate of hatching in the laboratory is between 30% and 82%, but it could be up to 96.53%. During the nymph stage, the mirid moves on the cocoa tree to feed of the pods and the shoots. The rate of survival of nymphs is around 68% [3]. The sex ratio is 1 female for 0.7 male. After the last phase of nymph. a female mirid (*S. singularis*) starts to mate after 6 to 10 days while the male mirid starts after 2 days [5]. After the mating, a female mirid is considered to be mature, and it can lay her eggs in many pods at different times. To lay an egg, a female mirid needs 4 to 8 days, on average, after the mating. More than 72.1% of female mirids become mature (able to lay eggs) [17].

In average, the fecundity of female mirids is 52.5 eggs, and the fecundity period lasts, on average, for 16 days [26]. The daily survival of mirids adult is around 98.14% for immature females, 92.8% for mature females, and 93% for males [26]. To obtain a new generation, 50 to 60 days are needed on average [17].

In order to understand and forecast the mirid population dynamics, several studies have been done using a variety of approaches and techniques : development of a rearing method [5], sheeting method [16], phe-

romone trapping [25, 17] and mathematical modelling [26]. Babin *et al.* [5] developed a rearing method for cocoa mirids in order to assess the performance using the calculation of life-table parameters and to investigate the relationship between rearing performance, especially female fecundity, and climatic conditions. Sarfo *et al.*, [25] studied the potential of sex pheromone trapping as an alternative environmentally-acceptable method for managing the mirids. Based on the behavioral responses of the mirids to pheromones in traps, parameters were standardized for efficient performance of the traps.

In general, field studies like those of Sarfo *et al.* [25] and Babin *et al.* [4] focus on finding correlations between observed data (mirid populations) and explanatory variables (shade, temperature, rain, pheromone, etc.). Linear regression helps to determine the relationship between variables. Such statistical models can help to explain or forecast relationships between parameters. The limits of such field studies are that they are expensive, and do not allow to test many hypotheses. Additionally, the quality of the acquired data is sometimes poor [17]. Moreover, the laboratory data and field data are not always accordance due to the fact that, in the field, it is plausible that not all parameters that influence mirids are considered. Field data are often site-specific, and extrapolation of their results to other regions is not obvious [10, 13]. In field studies, the combination of many parameters can also pose a severe challenge. A theoretical modelling approach or representation of such an environment could be useful to explore how the mirids live and which parameters are important to them.

Tapi *et al.* [26] developed a generic mathematical model for mirid populations to better predict its dynamics in a plot under different management strategies. Their first model [26] is a compartmental model with constant or periodic parameters. It is developed based on the mirid life cycle. They have done a theoretical and sensitivity analysis to provide numerical simulations. Following this, Tapi *et al.* [26] developed a second model (a time-delayed model) based on the previous one. The second model was better in terms of considering the developmental time at different stages. After a brief theoretical analysis, they provided numerical simulations with/without control efforts. Mathematical models allow predicting the long-term dynamics of the studied systems. However, the variability between individuals is difficult to represent [24]. Moreover, mirids live in traditional cocoa agro-forests which are complex systems where interactions and components are multiple and different [18]. Although, mathematical models proposed in the literature are easy to prove, they can be difficult to explain to non-mathematicians. Therefore, they are not the best tools to communicate with certain target groups, such as

farmers [19].

The design of each model is founded upon approximations of the pest's biology and behavior on a population level [13]. The models presented in the literature are often dedicated to specific farms, and local interactions of individual mirids are not represented or incorporated explicitly into the population dynamics. In order to better understand the mirid dynamics, it would be useful to consider the local interactions of mirids in space and time. Thus, an agent-based approach seems to be suitable to deal with mirid motion and local interactions. Agent-based modeling is the computational study of social agents interacting in an autonomous manner as evolving systems. It allows the study of complex adaptive systems and facilitates investigations into how macro-phenomena develops from micro-level behavior among heterogeneous sets of interacting agents [14]. Mirids are governed by their biological preferences and limitations as documented in the literature. Therefore, and by representing mirids as individual agents, the resulting relationships between these agents can be used to predict population dynamics of the pest. Additionally, this prediction is more realistic over space and time (based on local interactions) than the ones with the population aggregated mirid models.

In the context of our work where users are in the loop, we think that there is a need for a simplified way to explain the system results to users, and to interact with them and acquire feedback. In our opinion, simulation and visualization are a great way to explain the system and its results, while questionnaires and direct feedback are important to make sure the user in the loop understands the process.

2 Multi-agent Model of Mirids

The modelling of our system is based on the second version of GAIA proposed by Zambonelli *et al.* in [28]. They have extended the version of Wooldridge *et al.* [28, 27] towards social issues and open systems. GAIA prescribes following an ordered sequence of steps, an ordered identification of a set of organizational models, and an indication of the interrelationships guiding developers towards the development of a Multi-agent System (MAS). The GAIA methodology starts first with the analysis phase to collect and organize the specifications which are the basis for the design of the computational organization [28]. Then, the architectural design phase defines the most proper organizational structure for the MAS, the topology of interactions in the MAS, and the control regime of these interactions. Finally, the detailed design is responsible for identifying, eventually, the agent model and the services model, which act as guidelines for the actual implementation of agents and their activities. In order to emphasize the microscopic behavior of a

mirid and for reasons of simplicity, we assume that the mirids live and grow on one cocoa pod. However an extension of this work could consider several cocoa pods in a 3D dimension. A cocoa pod constitutes the environment like showed in Figure 1. In the mentioned figure, black points represent the places where mirids have fed themselves. Mirid's position is represented in a 2D plan. This paper focuses on the modeling of mirids interactions through the environment without considering external interactions such as humans.



FIGURE 1 – Mirid on a cocoa pod [5]

2.1 Analysis

Environment model : For simplicity reasons, we assume that the environment in our model is restricted to the cocoa pod where the mirids live, feed themselves, grow, mate and die. Table 1 presents the environment model for our work.

TABLE 1 – The environment model

Action	Environment Abstraction	Description
Reads	Agents position	The cocoa pod identifies position occupied by the mirid
Changes	Surface destroyed	Part of the surface of the pod is destroyed by the mirid
	Position updated	The mirid moved

TABLE 2 – Global variables of the environment

Variables	Description
Relative humidity	The climate gives the current relative humidity of the system
Temperature	The climate gives the current temperature of the system
Shade	The climate gives the current shade received by a part of the system

We have identified some global variables that influenced the dynamics of the system. Climate parameters and shade impact the growing of the mirid and the

TABLE 3 – The MIRID functional role schema

Role name : MIRID
Description : This role is in charge of representing the birth, growth, mating, feeding, and death of a cocoa mirid
Protocol and activities : move, feed, grow, be born, mate, lay egg, die
Permissions : Reads : PosX, PosY Changes : Occupied_area, an area of cocoa pod occupied by the mirid Destroyed_area, an area destroyed by the feeding of mirids
Responsibilities : Liveness : $MIRID = (to_be_born.grow + .move.feed.[mate].[lay_egg]).die)^n$ Safety : numberMirid > 2, miridMale.PosX=MiridFemale.PosX, miridMale.PosY=MiridFemale.PosY, New_Destination ∈ set_points_of_cocoa_pod, Content(New_Destination) = ∅

cocoa pod. Table 2 presents the description of these global variables.

Preliminary role model : The aim of this model is to describe the whole activity and procedure that can be done by a role. We identify a role in our system (MIRID). Table 3 presents the preliminary role model of our system (emphasized represented activities). To execute the MIRID role, it is important to know at which position the mirid is situated in the plan. Its actions can impact the environment by destroying a part of the pod or by occupying a set of points. A liveness property of the MIRID role specifies that a mirid, after its birth, must grow. After certain progression, it can start moving and feed itself. When it is mature, it can mate and if it is a female mirid it can lay eggs. At the end, it will die eventually. In our system, the role MIRID must guarantee that we have more than two mirids (at least both genders). For mating, female and male must be at the same position. To move in the system, there should be an empty position inside the whole system.

Preliminary interaction model : The preliminary interaction that can be observed here is the mating between a male mirid and a female one. This interaction starts with the stigmergy¹ communication. The

1. This term has been introduced by the French biologist Pierre-Paul Grassé in 1959 for describing the termite behavior. It is defined as : “Stimulation of the workers by the work that they performed.” This term expresses the notion that the ac-

female mirid emits a pheromone on the pod. When the male mirid perceives the pheromone, it will move to look for the female. When the position of the female mirid and the male mirid is the same, the mating starts. Figure 2 presents the preliminary interaction in our model.

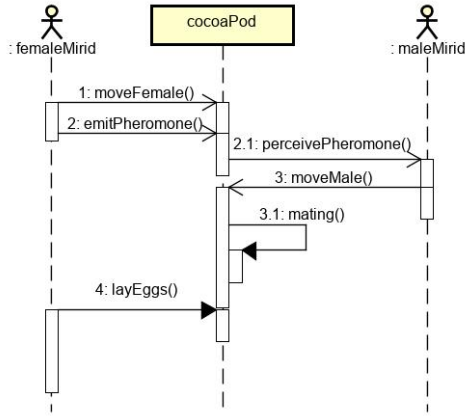


FIGURE 2 – The mating interaction protocol

Organizational rules : According to the authors of GAIA version 2 in [28], the main goal of organizational rules is to present general relationships between roles, between protocols, and between roles & protocols. Organizational rules have two properties : the *liveness property* which defines how the dynamics of the organization should evolve over time and the *safety property* which defines the time-independent global invariants of the organization that must be respected. The safety property of our system is to ensure the continuity of the mirid species. For this, the mirid role is played indefinitely. The liveness property of our system is to ensure that the population evolves over time. For this, the mirids must mate to create a new generation of mirids. Table 4 shows the different organizational rules of our system.

TABLE 4 – Organizational rules

Organizational Rules	Description
Liveness rules	
MIRID	To lay an egg or to grow, there is a need for mating
safety rules	
$Mirid^\omega$	The role MIRID can be played indefinitely

tions of an agent leave signs in the environment. These signs are perceived by itself and other agents, which determine their next actions [23]

2.2 Architectural Design

Organizational structure : From the biological description of the mirid on a cocoa pod, we define the organizational structure of our system. A peer relationship is the type of relations inside our system. It means that there is an indirect conflict on food. All agents of our system collaborate to keep the continuity of their population. Whatever the mirid (i or j) of our system, it collaborates with the others. The male and female mirids also collaborate (\overrightarrow{peer}). The lack of competition is due to the abundance of food. This can be simply expressed by Equation 1.

$$\forall(i, j) \in Mirid^2, i \neq j \Rightarrow i \overrightarrow{peer} j \quad (1)$$

Role model : After having the organizational structure, it is possible to complete the role model. This model presents the internal state of agents in the system. State-transition of the role is showed in Figure 3. It defines the specification of activities and services of the basic role and the organizational roles according to the above adopted organizational structure.

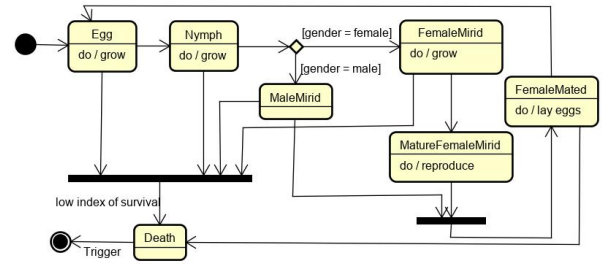


FIGURE 3 – State-Transition diagram of the role MIRID

Interaction model : The goal of this part is to identify new protocols or interactions deriving from the adopted organizational structure. Some interactions, notably *laying eggs* (Figure 4) and *feeding* (Figure 5), result from the organizational topology and the identification of agents.

2.3 Detailed Design

Agent model : Building an agent model consists, essentially, of establishing a link between the roles and the agent. It indicates how many instances of each agent will exist in the system, and which roles each agent will implement. It permits us to specify how many instances of each class will appear in the MAS using the cardinality standard [9]. Our system has only one role named MIRID which can be played once or many times by the agent MIRID. Figure 6 shows our agent model.

Service model : It identifies the services associated to each agent or class, with the role to be played by each agent class. Generally, the protocols, activities

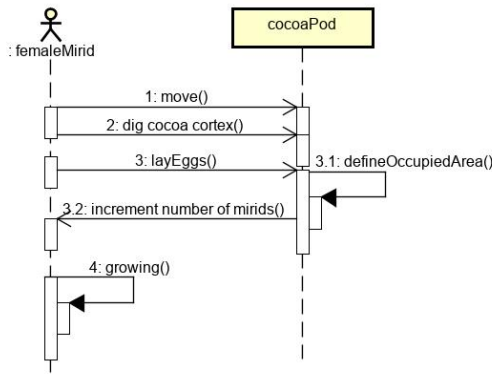


FIGURE 4 – The *lay eggs* interaction protocol

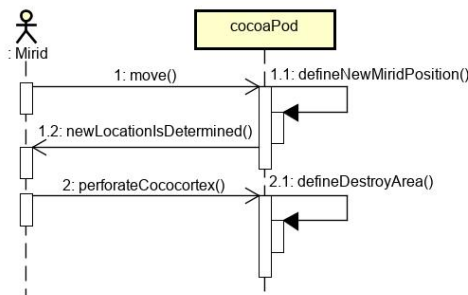


FIGURE 5 – The *feeding* interaction protocol

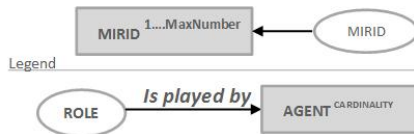


FIGURE 6 – Agent model

and liveness expressions of the role allow us to find different services that will be implemented by an agent. Agent performs at least one service, it is necessary to document the properties of services. Table 5 specifies our service model.

3 Discussion

The previous section presents our agent-based model of the mirid on a cocoa pod, which is a complex system. In order to implement such a complex system, agent-based simulation frameworks have been developed to allow transferring real-world scenarios into executable models (*i.e.* simulating the activities of mirids in a digital environment) [20]. To implement the model using agent-based simulation, some requirements should be respected :

- Specify the characteristics of our environment. The size of the pod cocoa depends on the cocoa variety and it extremely varies due to climate fac-

TABLE 5 – The service model

Services	Inputs	Pre-conditions	Post-conditions
Initialisation	Pod_area and Mirids number	Mirids number ≥ 2 and at least one female and one male	
Moving	Destination position	Destination position is free	Occupied position
Death	Actual position	Mirids number > 2	Mirids number decreased
Feeding	Feed area	\exists eatable area	Destroyed area

tors.

- Define the surface of the cocoa that should be considered according to the cocoa variety. The surface of the cocoa increase with time.
- Define the position of our initial mirid. When the sun appears, mirids concentrate on the shade-less areas, they disperse themselves when the night comes. For the moving of mirids, we shall define a set of points where the mirids can go to if they are empty. Mirids move only when the sun appears and they disappear for mating or if there is an enemy around them.
- For the reproduction, the female mirid emits a pheromone that will attract the male mirid.
- Define the probability of death of mirids at any stage of the development.

All those steps of implementation will help us implement our model using agent-based simulation. Our models, simulations and interfaces should be explainable to human users (entomologists, agricultural engineers, farmers and decision makers) in the cocoa domain. Explaining to users how intelligent systems come to their conclusions is an area of research with a history in expert systems and recommended systems [12, 15]. Recently, as intelligent systems are getting increasingly complex. eXplainable Artificial Intelligence (XAI) reemerged to push for interpreting the process of the machine learning mechanisms, and understanding the decisions of robots and agents [2, 8].

This work is on progress, and in the future we want to add the different actions of different human users (entomologists, agricultural engineers, farmers and decision makers). We need to measure how the regularity of their actions can impact the damage caused by mirids. Therefore, the main goal of our work is to provide tools (simulators and models) for the users. The tools make a recommendation, decision, or action. Then, they provide an explanation to the user that justifies its recommendation, decision, or action.

Entomologists will use these tools for the preparation of their experimentation on the fields. It will help them have an idea of the type of results that they could have. The usefulness of our tools for agricultural engineers and farmers is that it will help in the pest surveillance and the forecasting of optimal timing for insecticide spraying. Farmers of cocoa are mostly poor, so it is important to them to have optimal timing for insecticide spraying. At the end of our simulations, we could estimate the damage caused by the mirids to the cocoa pods. This type of information is very important for decision makers and the government.

Explainability has to be handled in two steps in our system (Figure 7) :

1. From the cocoa literature, some works have been done on the mirid, especially by entomologists. We use these works to build our agent-based models. After the modelling, implementation outputs are interpreted by graphs. At this step, we need the interaction of entomologists to validate our models. They will be able to evaluate if our results are close to what they often observe in the fields. With the feedback provided by the entomologists, the model can be revisited and better explainable interfaces can be built.
2. All outputs of previous step concern directly the farmers and decision makers. Interfaces are necessary because they allow individual tracking of each agent in the system. This individual follow-up would better explain emerging behaviors, which is not the case for macroscopic modeling. Interfaces are useful to forecast the future according to parameters that they would have entered previously. Farmers could also use those outputs to have an optimal plan for the different interventions (insecticide spray) on the field. Also the entomologist could need this outputs to have a first idea of the results that they could have for their experimentations.

The ultimate goal for this work will be to explain to users the behavior of an agent-model and simulation of the mirids in a cocoa farm. This explanation will help the users understand the behavior of the mirids, and hence, help in their decision making process to prevent the damage caused by mirids to the cocoa pods in cocoa farms.

4 Conclusion

In this paper, an initial agent-based model representing the population dynamics of mirids interacting on a cocoa pod is presented. The model is built based on the literature and by different steps of GAIA methodology.

According to our knowledge of the literature, our work is the first agent-based model that is developed to study mirids pest. This approach will greatly improve

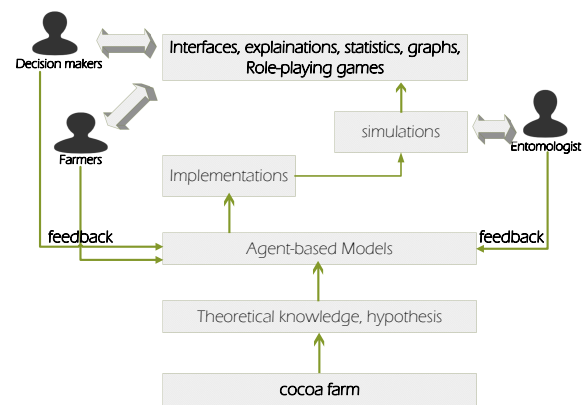


FIGURE 7 – Workflow of our future work

the prospects of understanding small scale interactions between individual components of cocoa farm.

In terms of integrating pest management strategy, our model aims to create a comprehensively design-specific, detailed, and integrated system to control specific pest species within specific crop zones. With enough refinement and insight, the behavior of mirids will possibly be evaluated at a specific zone. Next areas of our work could be : (i) the simulation and validation of our model; (ii) the evaluation of damage caused by the mirids; (iii) the integration of the impact of human actions (spread of insecticide) on the mirids population; (iv) how to auto-regulate the population of mirids by the use of ants for biological control of insects; (v) testing our model and simulation with entomological research data.

Acknowledgments

This work is supported by AUF(Agence Universitaire de la Francophonie), and partially by the Regional Council of Bourgogne Franche-Comté (RBFC, France) within the project UrbanFly 20174-06234/06242.

Références

- [1] J. C. Anikwe, A. A. Omoloye, P. O. Aikpokpodion, F. A. Okelana, and A. B. Eskes. Evaluation of resistance in selected cocoa genotypes to the brown cocoa mirid, *Sahlbergella singularis* Haglund in Nigeria. *Crop Protection*, 28(4) :350–355, 2009.
- [2] S. Anjomshoe, A. Najjar, D. Calvaresi, and K. Främling. Explainable agents and robots : Results from a systematic literature review. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1078–1088. International Foundation for Autonomous Agents and Multiagent Systems, 2019.
- [3] R. Babin. *Contribution à l'amélioration de la lutte contre le miride du cacaoyer Sahlbergella singularis* Hagl.(Hemiptera : Miridae). *Influence des fac-*

- teurs agro-écologiques sur la dynamique des populations du ravageur. PhD thesis, Université Paul Valéry-Montpellier III, 2009.
- [4] R. Babin, J. C. Anikwe, L. Dibog, and J. P. Lumaret. Effects of cocoa tree phenology and canopy microclimate on the performance of the mirid bug *Sahlbergella singularis*. *Entomologia Experimentalis et Applicata*, 141(1) :25–34, 2011.
- [5] R. Babin, D. H. Bisseleua, L. Dibog, and J. P. Lumaret. Rearing method and life-table data for the cocoa mirid bug *Sahlbergella singularis* Haglund (Hemiptera : Miridae). *Journal of Applied Entomology*, 132(5) :366–374, 2008.
- [6] R. Babin, C. Djieto-Lordon, C. Cilas, L. Dibog, R. Mahob, and C. F. B. Bilong. True Bug (Heteroptera) Impact on Cocoa Fruit Mortality and Productivity. *Journal of Economic Entomology*, 105(4) :1285–1292, 2012.
- [7] R. Babin, C. Fenouillet, T. Legavre, L. Blondin, C. Calatayud, A. M. Risterucci, and M. P. Chapuis. Isolation and characterization of twelve polymorphic microsatellite loci for the cocoa mirid bug *sahlbergella singularis*. *International Journal of Molecular Sciences*, 13(4) :4412–4417, 2012.
- [8] D. Calvaresi, Y. Mualla, A. Najjar, S. Galland, and M. Schumacher. Explainable multi-agent systems through blockchain technology. In *Proceedings of the 1st International Workshop on eXplanable TRansparent Autonomous Agents and Multi-Agent Systems (EXTRAAMAS 2019)*, 2019.
- [9] M. Cossentino, V. Hilaire, A. Molesini, and V. Seidita. *Handbook on agent-oriented design processes*. Springer, 2014.
- [10] C. M. Duarte, J. S. Amthor, D. L. DeAngelis, L. A. Joyce, R. Maranger, M. Pace, J. Pastor, and S. Running. The limits to models in ecology. *Models in ecosystem science*, pages 437–451, 2003.
- [11] A. Fountain and F. Hütz-Adams. Baromètre du cacao 2015. *Créat. Commons Licens. Attrib.-ShareAlike 4.0 Int*, 2015.
- [12] N. Gilbert. Explanation and dialogue. *The Knowledge Engineering Review*, 4(3) :235–247, 1989.
- [13] V. Grimm. Mathematical models and understanding in ecology. *Ecological modelling*, 75 :641–651, 1994.
- [14] M. A. Janssen. Internet encyclopedia of ecological economics : Agent-based modelling. 2005.
- [15] R. Kass, T. Finin, et al. The need for user models in generating expert system explanations. *International Journal of Expert Systems*, 1(4), 1988.
- [16] N. Kouamé, F. K. N’Guessan, H. A. N’Guessan, P. W. N’Guessan, and Y. Tano. Variations saisonnières des populations de mirides du cacao dans la région de l’indénié-djuablin en côte d’ivoire. *Journal of Applied Biosciences*, 83(1) :7595–7605, 2014.
- [17] R. J. Mahob, R. Babin, G. M. ten Hoopen, L. Dibog, Yede, D. R. Hall, and C. F. Bilong Bilong. Field evaluation of synthetic sex pheromone traps for the cocoa mirid *Sahlbergella singularis* (Hemiptera : Miridae). *Pest Management Science*, 67(6) :672–676, 2011.
- [18] E. Malézieux, Y. Crozat, C. Dupraz, M. Laurans, D. Makowski, H. Ozier-Lafontaine, B. Rapidel, S. De Tourdonnet, and M. Valantin-Morison. Mixing plant species in cropping systems : concepts, tools and models : a review. In *Sustainable agriculture*, pages 329–353. Springer, 2009.
- [19] M. Moglia, S. Cook, and J. McGregor. A review of Agent-Based Modelling of technology diffusion with special reference to residential energy efficiency. *Sustainable Cities and Society*, 31 :173–182, 2017.
- [20] Y. Mualla, W. Bai, S. Galland, and C. Nicolle. Comparison of agent-based simulation frameworks for unmanned aerial transportation applications. *Procedia Computer Science*, 130(C) :791–796, 2018.
- [21] M. A. Ngo Bieng, C. Gidoin, J. Avelino, C. Cilas, O. Deheuvels, and J. Wery. Diversity and spatial clustering of shade trees affect cacao yield and pathogen pressure in Costa Rican agroforests. *Basic and Applied Ecology*, 14(4) :329–336, 2013.
- [22] I. C. Organization. Icco quarterly bulletin of cocoa statistics. *The International Cocoa Organization*, XLII(3), 2016.
- [23] H. Parunak, S. A. Brueckner, R. Matthews, and J. Sauter. How to calm hyperactive agents. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 1092–1093. ACM, 2003.
- [24] N. Quang Huynh, T. Nguy en Huu, A. ud Grignard, H. X. Huynh, and A. Drog oul. Coupling equation based models and agent-based models : example of a multi-strains and switch SIR toy model. *EAI Endorsed Transactions on Context-aware Systems and Applications*, 4(11) :152334, 2017.
- [25] J. Sarfo, C. Campbell, and D. Hall. Design and placement of synthetic sex pheromone traps for cacao mirids in ghana. *International Journal of Tropical Insect Science*, 38(2) :122–131, 2018.
- [26] D. Tapi, B. Beihle, S. Bowong, and Y. Dumont. Models for Miridae, a cocoa insect pest. Application in control strategies. *Mathematical Methods in the Applied Sciences*, (March) :1–24, 2018.
- [27] M. Wooldridge, N. R. Jennings, and D. Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems*, 3(3) :285–312, 2000.
- [28] F. Zambonelli, N. R. Jennings, and M. Wooldridge. Developing multiagent systems : The gaia methodology. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 12(3) :317–370, 2003.

Apprentissage de relations floues pour l'annotation sémantique expliquée avec peu de données

Régis Pierrard^{1,2}Jean-Philippe Poli¹Céline Hudelot²¹ CEA, LIST, 91191 Gif-sur-Yvette cedex, France.² Mathématiques et Informatique pour la Complexité et les Systèmes, CentraleSupélec, Université Paris-Saclay, 91190, Gif-sur-Yvette, France.

{regis.pierrard, jean-philippe.poli}@cea.fr, celine.hudelot@centralesupelec.fr

Résumé

Malgré son récent succès, l'apprentissage profond semble loin d'égaliser certaines capacités humaines comme l'apprentissage à partir de peu d'exemples, le raisonnement ou encore l'explicabilité. Dans cet article, nous mettons l'accent sur l'annotation d'objets dans des images et nous présentons un cadre de raisonnement s'appuyant sur l'apprentissage de relations floues qui permet notamment de générer des explications. Étant donné un ensemble de relations, les plus pertinentes d'entre elles sont induites et combinées pour construire des contraintes permettant de définir un problème de satisfaction de contraintes floues. La résolution de ce problème nous permet d'annoter l'image et de générer des explications à l'aide des contraintes utilisées. Nous testons notre approche sur un jeu de données public avec deux objectifs : effectuer une annotation d'organes et fournir une explication à chaque annotation. Nous montrons que le modèle que nous obtenons peut générer des explications tout en atteignant des performances élevées malgré un jeu de données pouvant être constitué de seulement deux exemples.

Mots Clef

Explicabilité, annotation, logique floue, peu de données.

Abstract

Despite the recent successes of deep learning, such models are still far from some human abilities like learning from few examples, reasoning and explaining their decisions. In this paper, we focus on object annotation in images and we introduce a reasoning framework that is based on learning fuzzy relations on a dataset for generating explanations. Given a catalogue of relations, it efficiently induces the most relevant relations and combines them for building constraints in order to both solve the target task and generate explanations. We test our approach on a publicly available dataset on which the goal was both to perform multiple organ annotation and to provide explanations. We show that our model can generate explanations and achieve high performance despite being trained on a

small dataset containing as few as a couple of examples.

Keywords

Explainability, annotation, fuzzy logic, few-data learning.

1 Introduction

Au cours de ces dernières années, l'explicabilité est devenue un enjeu majeur pour les intelligences artificielles [12, 17]. Cela fait notamment écho à la popularité des réseaux de neurones profonds, qui peuvent atteindre des performances très élevées dans de nombreuses tâches mais ne sont pas adaptés à la génération d'explication [14, 27]. Être capable d'expliquer les résultats produits par les intelligences artificielles est utile non seulement pour comprendre leur raisonnement, mais aussi pour les rendre plus crédibles et dignes de confiance aux yeux des utilisateurs [34]. Dans des domaines en lien direct avec l'être humain tels que l'analyse d'images médicales [25] ou les voitures autonomes, on ne peut pas prendre de décisions en faisant une confiance aveugle à un modèle car les conséquences peuvent être désastreuses.

Plusieurs définitions d'*interprétabilité* et d'*explicabilité* ont été données dans la littérature [10, 16, 24, 29], mais aucun consensus ne se dégage et ces deux notions sont mêmes parfois utilisées de manière interchangeable. Globalement, il émerge que l'interprétabilité est la capacité à présenter des informations sur le fonctionnement du système en des termes intelligibles, alors que l'explicabilité est la capacité à décrire de manière précise et logique le fonctionnement d'un système. Dans cet article, nous nous focalisons sur l'explicabilité et donc sur la restitution du raisonnement effectué par notre modèle. Pour obtenir des explications, une première famille de méthodes consiste à apprendre une approximation locale et interprétable autour de la prédiction fournie par un modèle de type boîte noire [26, 34]. Ces approches peuvent traiter tout type de modèle et sont donc adaptées aux réseaux de neurones profonds. Cependant, bien qu'elles soient capables d'extraire les caractéristiques principales qui ont servi à générer une prédiction, elles ne peuvent pas exactement reconstituer le

raisonnement effectué par le modèle cible. La seconde famille d'approches repose sur l'utilisation de modèles qui sont propices à la génération d'explications, tels que les arbres de décisions, les règles de décision ou l'obtention d'un modèle explicable à partir d'un modèle de type boîte noire par distillation [18]. Leur avantage principal est qu'ils permettent de facilement reproduire le raisonnement qui a mené à un résultat donné, ce qui est très utile pour la génération d'explications. Cependant, ces modèles risquent de ne pas être aussi performants que des modèles de type boîte noire car l'explicabilité a souvent un coût. En effet, il existe un compromis entre performance et explicabilité [17]. Dans ce papier, nous proposons une approche basée sur cette seconde famille de méthodes qui compense ce compromis en n'apprenant que sur une faible quantité de données étiquetées dont l'acquisition est coûteuse. Cela repose notamment sur des études cognitives sur l'interprétation humaine des images : (1) l'importance des relations spatiales et contextuelles en reconnaissance d'objets et de scènes [1], (2) la capacité des humains à n'avoir besoin que de quelques exemples pour apprendre à accomplir une tâche [23, 37]. Ce second point est notamment abordé par les zero-shot [21], one-shot et few-shot learning [13]. Ils visent à effectuer cela en utilisant les précédents exemples dans la base d'apprentissage ou les représentations disponibles. Cependant, de telles méthodes nécessitent toujours d'avoir des informations entre les classes connues et les classes inconnues [39].

Notre objectif est de proposer une nouvelle approche qui permet d'apprendre à raisonner et de générer à la fois des annotations et des explications à partir d'un faible nombre d'exemples d'apprentissage. Elle s'appuie notamment sur l'utilisation de relations floues, qui permettent de prendre en compte à la fois des informations quantitatives et des informations qualitatives. Pour un exemple donné, le système cherche les régions de l'image qui satisfont au mieux les relations entre les objets d'intérêt. Nous modélisons cela sous la forme d'un problème de satisfaction de contraintes. Dans la section 3, nous décrivons la démarche globale qui consiste en trois étapes principales : l'évaluation des relations, l'extraction des relations les plus pertinentes et la génération des contraintes pour résoudre un problème de satisfaction de contraintes et produire des explications. Dans la section 4, l'approche proposée est évaluée sur un exemple d'annotation d'images médicales. Ceci est un bon exemple de raisonnement spatial car la reconnaissance des différents organes repose essentiellement sur leur disposition spatiale. De plus, l'utilisation d'un jeu de données médicales présente plusieurs défis, tels que sa faible taille et l'importance des explications dans ce domaine. Nous avons testé et comparé notre modèle à l'état de l'art et nous avons montré que notre approche est capable d'obtenir de bonnes performances et de générer des explications malgré un faible nombre d'exemples d'apprentissage.

2 Rappels

2.1 Logique floue

La logique floue et la théorie des ensembles flous ont été introduits par Zadeh [41]. Il s'agit d'une extension de la logique Booléenne qui permet notamment de gérer l'imprécision des données. Alors qu'une variable est soit vraie soit fausse en logique Booléenne, elle peut prendre n'importe quelle valeur entre 0 (fausse) et 1 (vraie) en logique floue. Cette gamme de valeurs, qu'on appelle degrés, sert à gérer l'imprécision. Dans un univers A , un ensemble flou F est défini par une fonction d'appartenance $\mu_F : A \rightarrow [0; 1]$. Cette fonction, intitulée *fonction d'appartenance de F* , représente dans quelle mesure chaque $a \in A$ appartient à F . Si F est un ensemble non-flou, alors $\mu_F(a)$ vaut soit 0 quand a n'appartient pas à F , soit 1 si a appartient à F .

Il est également possible d'exprimer des relations entre deux univers différents. Pour deux univers A et B , une relation floue dyadique \mathcal{R} est un ensemble flou défini par une fonction d'appartenance $\mu_{\mathcal{R}} : A \times B \rightarrow [0, 1]$. Cette fonction attribue à chaque paire $(a, b) \in A \times B$ un degré d'appartenance à la relation \mathcal{R} . Les relations floues n -aire sont définies de la même façon. Un ensemble flou peut caractériser une expression du langage naturel lorsqu'il est associé à une description linguistique, ce qui est courant en logique floue.

2.2 Problème de satisfaction de contraintes floues

Un problème de satisfaction de contraintes est un problème dans lequel l'objectif est d'attribuer des valeurs à un ensemble de variables qui doivent vérifier un ensemble de contraintes. Les problèmes de planification en sont un exemple [30]. [11] en présente une extension au cadre de la logique floue afin de pouvoir gérer des paramètres et des contraintes flous lors de la résolution de problèmes de satisfaction de contraintes floues. Un tel problème est défini par un ensemble de variables X , un ensemble de domaines D et un ensemble de contraintes floues C . Par exemple, dans le cadre de l'annotation d'images, X correspond aux étiquettes que l'on souhaite attribuer aux objets de l'image, D représente les différentes régions à annoter et C est un ensemble de relations floues liant ces objets. Les problèmes de satisfaction de contraintes floues nous intéressent tout particulièrement car ils nous permettent d'effectuer l'annotation (en résolvant le problème) ainsi que de générer des explications à partir des contraintes utilisées.

Pour résoudre un tel problème, on utilise un algorithme de backtracking pour parcourir l'ensemble des solutions possibles. Avant chaque assignation d'une valeur à une variable, l'algorithme FAC-3 [11, 38] est utilisé pour maintenir la cohérence de l'ensemble des valeurs possibles. Cette opération de propagation des contraintes permet de réduire la taille de l'espace de recherche. À la fin, la solution retenue est celle qui vérifie au mieux les contraintes.

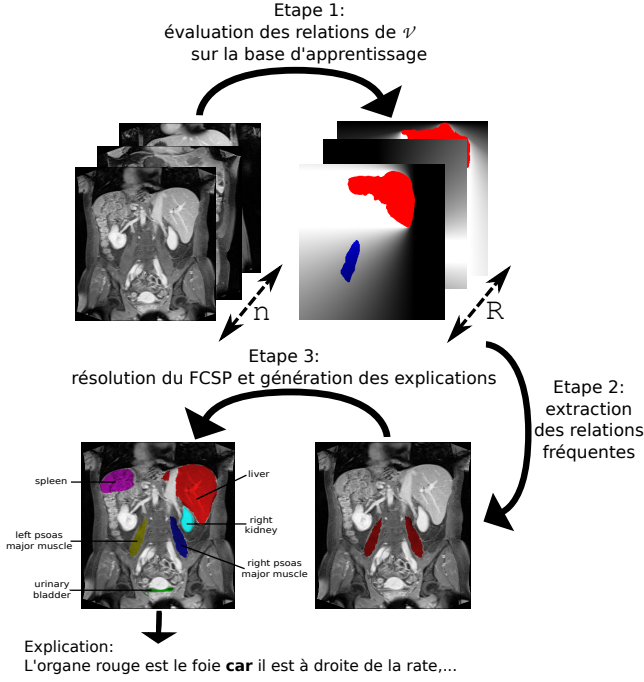


FIGURE 1 – Représentation de notre approche qui est constituée de trois étapes. Dans l'étape 1, les n_r relations floues du vocabulaire \mathcal{V} sont évaluées sur le jeu de données \mathcal{T} qui est composé de n images. Ensuite, les relations les plus fréquentes sont extraites sur la base d'apprentissage \mathcal{T}_{app} et elles sont utilisées pour déterminer les contraintes de notre problème. Enfin, un problème de satisfaction de contraintes floues est défini et résolu afin d'étiqueter les objets cibles pour chaque image de $\mathcal{T}_{\text{test}}$. Une explication s'appuyant sur les contraintes utilisées est fournie pour chaque labellisation d'objet.

3 Approche proposée

Dans cette partie, nous décrivons notre approche dont l'objectif est d'annoter les régions d'intérêt et de fournir une explication pour chaque annotation. Elle se divise en trois étapes : l'évaluation des relations entre les régions d'intérêt (Sec. 3.1), l'apprentissage des relations les plus pertinentes (Sec. 3.2) et la résolution d'un FCSP pour trouver les régions qui satisfont au mieux les relations apprises tout en générant des explications à partir des contraintes (Sec. 3.3). L'approche globale est représentée sur la Figure 1.

3.1 Étape 1 : évaluation des relations

L'objectif de cette première étape est d'évaluer différentes relations entre les régions d'intérêt (les organes par exemple) afin de pouvoir plus tard (dans l'étape suivante) déterminer celles qui sont les plus pertinentes.

Soit un ensemble d'apprentissage $\mathcal{T}_{\text{train}}$ composé de n images $\{\mathbf{i}_1, \dots, \mathbf{i}_n\}$ et un ensemble d'étiquettes \mathcal{Y} composé de N éléments $\{y_1, \dots, y_N\}$ tels que chaque image $\mathbf{i} \in \mathcal{T}_{\text{train}}$ est divisée en un ensemble \mathcal{O} de N régions d'intérêt $\mathcal{O} = \{o_{i,1}, \dots, o_{i,K}\}$ qui sont associées aux étiquettes

par la fonction suivante :

$$f : \mathcal{Y} \rightarrow \mathcal{O} \\ y_i \mapsto o_j \quad (1)$$

Soit $\mathcal{V} = \{\mathcal{R}_1, \dots, \mathcal{R}_R\}$ un ensemble de R relations. Nous appelons cet ensemble un *vocabulaire*. Il est défini par un expert et contient des relations qui sont potentiellement d'intérêt. Par exemple, il peut s'agir de relations directionnelles, comme *à gauche de*, ou de relations de distance, comme *près de*. Un vocabulaire plus riche devrait rendre le système plus expressif et améliorer les annotations et explications produites. Les relations de notre vocabulaire \mathcal{V} sont automatiquement évaluées sur les régions d'intérêt de chaque image de $\mathcal{T}_{\text{train}}$. La manière dont elles sont évaluées dépend de leur définition (cf. Sec. 4.2).

Pour chaque relation $\mathcal{R} \in \mathcal{V}$, nous appelons $\alpha(\mathcal{R})$ son arité. \mathcal{R} est évaluée pour tous les $\alpha(\mathcal{R})$ -uplet de régions d'intérêt possibles. Il est important de distinguer le vocabulaire \mathcal{V} de l'ensemble $\mathcal{V}_{\mathcal{Y}}$ des relations à évaluer entre les régions d'intérêt. Par exemple, pour une relation binaire $\mathcal{R} \in \mathcal{V}$, il faut évaluer l'ensemble $\{\mathcal{R}(f(y), f(y')) \mid y, y' \in \mathcal{Y}, y \neq y'\}$. Ainsi, le nombre d'évaluations à effectuer est :

$$\text{Card}(\mathcal{V}_{\mathcal{Y}}) = \sum_{j=1}^R \frac{N!}{(N - \alpha(\mathcal{R}_j))!} \quad (2)$$

À la fin de cette étape, nous avons donc une base de données $\mathcal{B} : \mathcal{T}_{\text{train}} \times \mathcal{V}_{\mathcal{Y}} \rightarrow [0; 1]$ de taille $n \times \text{Card}(\mathcal{V}_{\mathcal{Y}})$.

3.2 Étape 2 : apprentissage des relations pertinentes

Lors de cette étape, l'objectif est d'extraire les relations les plus pertinentes parmi celles évaluées à l'étape précédente. Pour une étiquette $y \in \mathcal{Y}$, nous faisons le postulat que les relations pertinentes impliquant les régions étiquetées comme y sont les plus fréquentes car elles devraient être vérifiées par la plupart, si ce n'est tous, des exemples associés à cette classe. Ainsi, l'apprentissage des relations les plus pertinentes revient à extraire les relations les plus fréquentes. Cette opération est effectuée classe par classe, il s'agit d'une stratégie *one vs all*. En effet, les relations pertinentes pour une classe d'objet ne le sont pas forcément pour une autre classe. Au sein d'une même classe, les exemples devraient être fortement corrélés entre eux. C'est pourquoi nous utilisons un algorithme d'extraction de motifs fréquents qui tire profit de cette propriété [32].

Un sous-ensemble J de relations est un ensemble appartenant à $2^{\mathcal{V}_{\mathcal{Y}}}$. Soit $J = \{r_1, \dots, r_p \mid r_j \in \mathcal{V}_{\mathcal{Y}}, 1 \leq j \leq p \leq \text{Card}(\mathcal{V}_{\mathcal{Y}})\}$. La fréquence de J dans la base de données \mathcal{B} s'appelle le *support* et est défini par :

$$\text{support}(J) = \frac{1}{n} \sum_{k=1}^n \min(\mathcal{B}(\mathbf{i}_k, r_1), \dots, \mathcal{B}(\mathbf{i}_k, r_p)) \quad (3)$$

Étant donné un seuil, J est dit *fréquent* si, et seulement si, son support est supérieur à ce seuil. L'algorithme que nous

utilisons s'appuie sur un opérateur de fermeture que nous définissons comme $h : 2^{\mathcal{V}} \rightarrow 2^{\mathcal{V}}$. Une première phase consiste à déterminer l'ensemble des sous-ensembles de relations vérifiant $h(J) = J$. De tels sous-ensembles de relations sont dits *fermés*. Ensuite, parmi les ensembles fermés, seuls ceux qui sont fréquents sont conservés. Enfin, l'ensemble des sous-ensembles fermés et fréquents de relations permet d'engendrer l'ensemble des sous-ensembles fréquents de relations.

Puisque nous suivons une approche one-vs-all, chaque classe d'objets y a son propre seuil de fréquence S_y dont la valeur est déterminée lors d'une phase de validation. La valeur de ce seuil a une influence directe sur le nombre d'ensembles de relations fréquents qui sont extraits. Si elle est trop élevée, alors pas ou peu d'ensembles de relations seront considérés comme fréquents. On risque alors de ne pas pouvoir discriminer les différentes classes et on se retrouve donc dans une situation d'*underfitting*. Au contraire, si la valeur du seuil est trop basse, alors des ensembles de relations non pertinents seront appris. Ceci mènerait à une situation d'*overfitting*.

À la fin de cette étape, pour chaque étiquette $y \in \mathcal{Y}$, nous avons un ensemble F_y de sous-ensembles fréquents de relations tel que $F_y \subseteq 2^{2^{\mathcal{V}}}$.

3.3 Étape 3 : résolution du problème de satisfaction de contraintes floues et génération des explications

Pour un exemple de test **i**, on peut obtenir par segmentation un ensemble de régions potentiellement d'intérêt. L'objectif de cette étape est de déterminer les étiquettes des régions qui satisfont au mieux les relations apprises à l'étape précédente. Cela peut être modélisé par un problème de satisfaction de contraintes floues. De plus, puisque les relations sont associées à une description linguistique, on peut générer une explication pour chaque annotation.

Pour chaque étiquette $y \in \mathcal{Y}$, nous avons obtenu à l'issue de l'étape précédente un ensemble F_y . Soit F_y^{\max} un ensemble défini de la façon suivante :

$$F_y^{\max} = \{z \in F_y \mid \text{Card}(z) = \max_{v \in F_y} (\text{Card}(v))\} \quad (4)$$

F_y^{\max} correspond donc à l'ensemble des sous-ensembles fréquents de relations de taille maximale. Chaque relation $\mathcal{R}(f(y), f(y'))$ présente dans les sous-ensembles de relations de $\bigcup_{y \in \mathcal{Y}} F_y^{\max}$ est directement convertie en une contrainte floue $c_{\mathcal{R}}(f(y), f(y'))$. Nous pouvons donc désormais construire un modèle défini par les contraintes qui ont été apprises ainsi que par les valeurs des seuils de fréquence. Reposant sur l'apprentissage de symboles (les relations) et n'ayant pas recours à la minimisation d'une fonction de perte, notre modèle est adapté à des jeux de données de faible taille.

Après segmentation, l'exemple de test **i** est divisé en K régions potentiellement d'intérêt $\{o_{i,1}, \dots, o_{i,K}\}$. Le pro-

blème de satisfaction de contraintes floues que nous cherchons à résoudre est donc le suivant :

$$X = \{x_y \mid y \in \mathcal{Y}\} \quad (5)$$

$$D = \{D_y \mid y \in \mathcal{Y} \text{ and } D_y = \{o_{i,1}, \dots, o_{i,K}\}\} \quad (6)$$

$$C = \{c_{\mathcal{R}}(x_y, x_{y'}) \mid \mathcal{R}(f(y), f(y')) \in U \text{ such as } U \subseteq \bigcup_{y \in \mathcal{Y}} F_y^{\max}\} \quad (7)$$

Chaque élément D_y de D est l'ensemble de toutes les régions qui pourraient être associées à la variable x_y . Au début, tous les D_y sont identiques. Ils sont ensuite modifiés par l'algorithme FAC-3 qui écarte les solutions incohérentes. Chaque contrainte de C ayant été évaluée, le problème peut alors être résolu et les étiquettes correspondant à chaque variable déterminées. Ensuite, pour chaque variable $x_y \in X$, une explication est générée en utilisant les contraintes obtenues à partir de F_y^{\max} . Par exemple, la contrainte $c_{\mathcal{R} \text{ à gauche de } (x_W, x_Z)}$ engendre "W est à gauche de Z". Ainsi, l'utilisation de contraintes générées à partir de F_y^{\max} permet d'exprimer une explication sous la forme : "classe y CAR cause₁, ..., cause_n". Pour une classe d'objets donnée, la contrainte la moins satisfaite nous donne un facteur de certitude pour modérer l'explication [5], e.g. "L'annotation de cet organe est *probablement* le foie...". Les contraintes apprises et ce facteur de certitude sont envoyés à un *surface realiser* comme simpleNLG [15] afin de les agréger en une phrase syntaxiquement correcte. L'évaluation de la qualité d'une explication est une tâche délicate. Une bonne explication est une explication cohérente [29], simple et générique [33], utile et pertinente [28]. Ces différents critères sont difficiles à évaluer de manière automatique et ils dépendent de la connaissance et des attentes de l'utilisateur. Trois types de solution sont proposés dans [10] : faire évaluer les explications par un expert, poser plusieurs questions simples à un groupe de personnes n'étant pas expert dans le domaine ciblé ou utiliser un modèle proxy dont l'explicabilité a déjà été évaluée pour juger l'explicabilité du modèle en cours d'évaluation.

4 Expériences

Dans cette partie, nous décrivons les expériences qui ont été menées sur un jeu d'images médicales. L'objectif est d'effectuer une annotation d'organes justifiée en apprenant un modèle à partir d'un faible nombre de données. Tandis que la détection d'organes est un sujet qui a été régulièrement abordé dans la littérature [36, 9, 31, 22], l'annotation d'organes n'a été traitée que dans [40]. Cette dernière méthode consiste à trouver dans le jeu de données des images qui ont des caractéristiques visuelles similaires à l'image étudiée et à l'étiqueter de la même manière que celles-ci. [22] aborde le problème de la détection d'organes sur des coupes abdominales. Leur méthode s'appuie sur des règles spatiales floues. Cependant, ces règles sont trop spécifiques et ne sont pas adaptées à d'autres jeux de données.

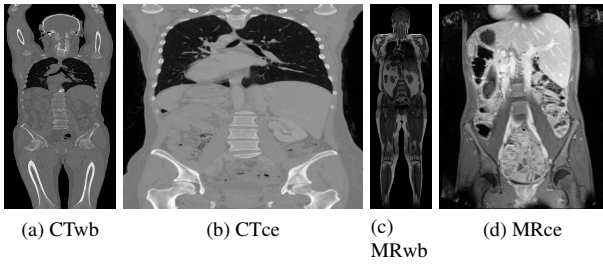


FIGURE 2 – Exemples correspondant aux quatre types d’images composant le jeu de données utilisé.

La classification d’organes a été traitée par [35] à l’aide d’un réseau de neurones convolutionnel. Leur méthode repose également sur des techniques d’augmentation de données permettant de répondre au problème de la faible taille du jeu de données utilisé.

4.1 Jeu de données

Partant du constat qu’il n’existe actuellement pas de jeu de données dédié à l’évaluation de l’explicabilité des modèles, nous avons utilisé un jeu de données initialement créé pour évaluer des méthodes de segmentation supervisée. Il s’intitule *Anatomy3* et a été présenté dans [19]. Il contient 391 images provenant de deux modalités différentes (CT et IRM) ainsi que les segments correspondant à certains organes de ces images. Certaines images représentent le corps entier (CTwb et MRwb) et les autres l’abdomen (CTce et MRce). Ce sont des images à trois dimensions obtenues par superposition de tranches en deux dimensions. Puisque nous travaillons sur des données à deux dimensions, nous avons sélectionné une tranche 2D par patient afin de construire un jeu d’images à deux dimensions. La figure 2 montre quatre tranches, chacune correspondant à un des quatre types d’images (CTwb, MRwb, CTce et MRce).

L’ensemble \mathcal{Y} d’organes auquel nous nous intéressons est composé du *foie*, de la *rate*, de la *vessie*, des *reins gauche et droit*, des *poumons gauche et droit* et des *muscles psoas gauche et droit*. Nous avons conservé toutes les images contenant ces neuf organes et leurs segments correspondants, ce qui nous donne un jeu de 35 images et 315 segments.

4.2 Paramètres expérimentaux

Apprentissage du modèle. Le modèle que nous construisons repose sur l’extraction des sous-ensembles de relations fréquents. Il y a autant d’hyperparamètres que d’organes. Ils correspondent tous au seuil utilisé pour juger la fréquence d’une relation. Afin d’optimiser la recherche de ces hyperparamètres, nous avons utilisé une validation croisée imbriquée [6] : (1) une validation croisée externe est réalisée pour obtenir un ensemble d’apprentissage et un ensemble de test à chaque itération, (2) une validation croisée interne est menée sur chaque base d’apprentissage

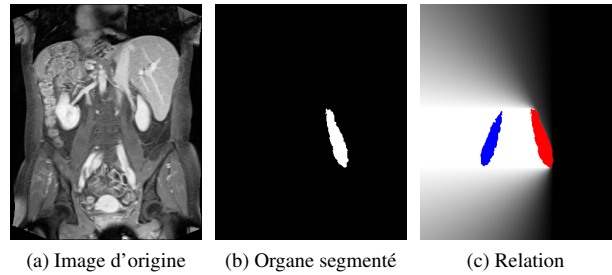


FIGURE 3 – (Affichage optimal en couleur) Exemple d’évaluation d’une relation. L’objectif est de déterminer la relation *l’organe bleu à gauche de l’organe rouge*. Étant donné une image en entrée (3a), un organe segmenté (3b) est considéré comme l’objet de référence. Cet organe est utilisé pour générer un paysage flou (3c) qui représente le degré d’appartenance de chaque pixel à la relation *à gauche de l’organe rouge*. La relation est déterminée en évaluant le degré d’intersection entre le paysage flou et l’organe bleu.

de la validation croisée externe pour obtenir un ensemble d’apprentissage interne et un ensemble de validation permettant d’ajuster la valeur des hyperparamètres. Cela permet d’obtenir une estimation de l’erreur non-biaisée par la construction de l’ensemble de validation.

Dans la boucle interne, l’optimisation des hyperparamètres est effectuée par optimisation bayésienne sur 20 itérations avec un processus Gaussien comme a priori. La fonction d’acquisition utilisée ici est l’*expected improvement* [20].

Relations. De nombreuses relations spatiales floues ont été étudiées dans la littérature [4]. Dans nos expériences, nous utilisons des relations directionnelles, de distance et de symétrie. Les relations directionnelles et de distance [2, 3] sont obtenues en générant un paysage flou et en utilisant une approche de type reconnaissance de forme floue [7]. Comme on peut le voir sur la figure 3, le paysage flou est obtenu par la dilatation floue d’un objet de référence par un élément structurant dont la définition détermine le type de relation à évaluer. Soit S l’espace de l’image. Soit A un objet de référence dans S et $\mu_{A,\mathcal{R}}$ la fonction d’appartenance associée au paysage flou représentant la relation \mathcal{R} dont l’objet de référence est A . Soit μ_B la fonction d’appartenance associée à un objet B dans S . La relation \mathcal{R} entre A et B est alors obtenue par évaluation du degré d’intersection flou μ_{int} [4] entre $\mu_{A,\mathcal{R}}$ et μ_B :

$$\mu_{\text{int}}(\mu_{A,\mathcal{R}}, \mu_B) = \frac{\sum_{x \in S} \min(\mu_{A,\mathcal{R}}(x), \mu_B(x))}{\min\left(\sum_{x \in S} \mu_{A,\mathcal{R}}(x), \sum_{x \in S} \mu_B(x)\right)} \quad (8)$$

Sur la figure 3, la relation \mathcal{R} est à *gauche de*, l’objet de référence A est l’organe rouge et l’objet B est l’organe bleu. Afin d’obtenir un vocabulaire fini, nous n’avons utilisé que certaines valeurs pour les paramètres de ces relations afin de n’exprimer que des relations telles que *au dessus de* ou

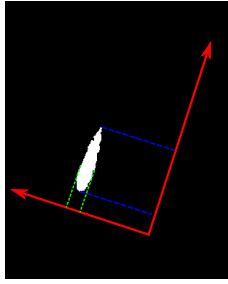


FIGURE 4 – (Affichage optimal en couleur) Illustration décrivant l'évaluation de la propriété *allongé*. Une analyse en composantes principales est effectuée afin d'obtenir les deux axes principaux. Ensuite, l'organe étudié est projeté sur ces deux axes. Soit p_g la projection verte (la plus petite) et p_b la projection bleue (la plus grande). Le degré d'appartenance est déterminé par $1 - \frac{p_g}{p_b}$.

près de.

La relation de symétrie axiale [8] que nous utilisons repose sur la détermination de l'axe qui maximise une mesure de symétrie entre deux organes. Puisque cette mesure n'est pas différentiable, ce problème d'optimisation est résolu à l'aide d'une méthode telle que le *downhill simplex*. Nous utilisons également une propriété. Elle peut être vue comme une relation unaire puisqu'elle ne caractérise qu'un seul organe. Cette propriété évalue à quel point la forme d'un organe est allongée. Étant donné un organe segmenté, une analyse en composantes principales est réalisée afin d'en obtenir les deux axes principaux. Ensuite, comme le décrit la figure 4, l'organe est projeté sur chacun de ces deux axes. Le rapport entre ces deux projections est alors utilisé pour déterminer le degré d'appartenance à cette propriété. Cependant, cette définition ne permet pas de bien juger l'allongement des objets concaves.

Notre vocabulaire de relations \mathcal{V} contient les relations suivantes : *à gauche de*, *à droite de*, *en dessous de*, *au-dessus de*, *près de*, *symétrique à* et *est allongé*. \mathcal{V} comporte donc six relations binaires et une relation unaire. Puisque $\text{Card}(\mathcal{V}) = 9$, le nombre de relations à évaluer pour chaque image est $\text{Card}(\mathcal{V}_y) = 441$. Cela contribue à rendre notre modèle expressif, mais le temps de calcul nécessaire à l'évaluation de toutes ces relations pour chaque image peut être important.

4.3 Initialisation du problème

Comme nous l'avons montré dans la section 3, la méthode complète est composée de trois étapes. Notre jeu de données contient pour chaque image les segments associés à chaque organe. Ces segments ne sont pas flous, mais la méthode resterait inchangée s'ils l'étaient.

L'objectif intermédiaire est de générer des contraintes afin de définir un problème de satisfaction de contraintes floues pour chaque exemple à évaluer. Une fois ce problème résolu, nous obtenons les segments correspondant à chaque organe et les contraintes peuvent désormais être utilisées

Organe	Valeur du seuil
Foie	0.96
Rate	0.86
Vessie	0.80
Rein droit	0.92
Rein gauche	0.89
Poumon droit	0.98
Poumon gauche	0.97
Muscle psoas droit	0.92
Muscle psoas gauche	0.88

TABLE 1 – Valeurs des seuils correspondant à chaque organe. Ce sont les hyperparamètres de notre modèle.

pour générer des explications.

L'ensemble de variables X et l'ensemble de domaines D de notre problème de satisfaction de contraintes sont les suivants :

$$X = \{x_{\text{foie}}, x_{\text{rate}}, x_{\text{vessie}}, x_{\text{rein}_d}, x_{\text{rein}_g}, x_{\text{poumon}_d}, x_{\text{poumon}_g}, x_{\text{psoas}_d}, x_{\text{psoas}_g}\}$$

$$D = \{D_{\text{foie}}, D_{\text{rate}}, D_{\text{vessie}}, D_{\text{rein}_d}, D_{\text{rein}_g}, D_{\text{poumon}_d}, D_{\text{poumon}_g}, D_{\text{psoas}_d}, D_{\text{psoas}_g}\}$$

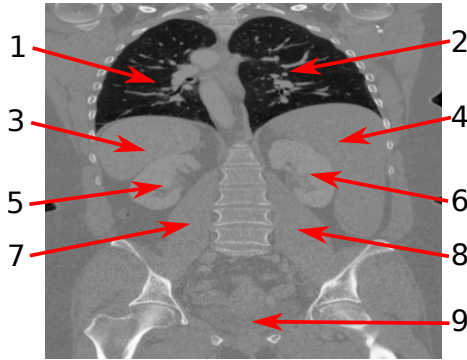
avec D_i l'ensemble des segments possibles pour la variable x_i . Il s'agit des segments fournis dans le jeu de données. Pour chaque organe $y \in \mathcal{Y}$, les contraintes floues sont générées à partir de l'ensemble des sous-ensembles fréquents de relations de taille maximale F_y^{max} . De plus, comme chaque organe est unique, il ne peut pas y avoir deux annotations identiques sur la même image. Il faut donc compléter C avec des contraintes imposant la différence entre chaque variable deux à deux. C'est pourquoi nous ajoutons la contrainte globale *AllDifferent*.

La définition du problème de satisfaction de contraintes floues est donc effectuée automatiquement. Ensuite, pour un exemple donné, le problème est résolu comme décrit dans la section 2.2.

4.4 Résultats

Un exemple est présenté sur la figure 5. Il y a neuf organes à annoter et donc neuf explications à fournir.

La mesure d'évaluation de notre modèle est la *précision*. Il s'agit du rapport entre le nombre d'annotations correctes pour tous les organes et le nombre total d'annotations effectuées. Nous obtenons une précision de 100% avec un modèle ne contenant que des relations directionnelles. La validation croisée externe contient trois plis (23/24 exemples dans la base d'apprentissage et 12/11 dans la base de test à chaque itération). La validation croisée interne comporte quatre plis. Comme il y a neuf organes à annoter, il y a neuf hyperparamètres à déterminer pour extraire les ensembles de relations fréquents. Les valeurs



L'annotation de l'organe 1 est très probablement le poumon gauche **car** il est à gauche du poumon droit (organe 2), il est symétrique au poumon droit et il est au-dessus de la rate (3).

L'annotation de l'organe 9 est probablement la vessie **car** elle est allongée, elle est en dessous du rein droit (6) et en dessous du rein gauche (5).

L'annotation de l'organe 4 est le foie **car**...

FIGURE 5 – Exemple d'annotations justifiées.

obtenues pour ces hyperparamètres sont affichées dans la table 1. Des contraintes pourraient être ajoutées sur les valeurs que peuvent prendre ces hyperparamètres pour spécifier la longueur des explications.

Nous observons que les explications reprennent bien les relations qui ont été extraites puis traduites en contraintes. Pour l'exemple proposé sur la figure 5, l'ensemble des contraintes associées au rein droit est :

$$C_{\text{rein}_d} = \{(x_{\text{rein}_d}, x_{\text{rein}_g}, R_{\text{symétrique à}}), (x_{\text{poumon}_d}, x_{\text{rein}_d}, R_{\text{au-dessus de}}), (x_{\text{rein}_d}, x_{\text{foie}}, R_{\text{à gauche de}}), (x_{\text{vessie}}, x_{\text{rein}_d}, R_{\text{en dessous de}}), (x_{\text{rein}_d}, x_{\text{rein}_g}, R_{\text{à droite de}}), (x_{\text{rein}_g}, x_{\text{rein}_d}, R_{\text{à gauche de}})\}.$$

Certaines de ces contraintes peuvent sembler redondantes, comme les deux dernières par exemple. Ce genre de situation peut arriver car le résultat d'une dilatation floue dépend de la forme de l'objet de référence. Comme deux organes différents ne peuvent pas avoir exactement la même forme, il peut y avoir de légères différences entre deux contraintes réciproques. Chaque organe est lié à un tel ensemble de contraintes. C est l'union de tous ces ensembles.

Bien que le modèle entraîné avec uniquement des relations directionnelles n'ait pas besoin d'autres types de relation pour obtenir une précision de 100%, ajouter de nouvelles relations peut permettre de construire de meilleures explications. Comme on peut le voir sur la figure 5, la relation *symétrique à* est utilisée dans l'explication de l'annotation du poumon gauche. Cette relation n'est pas nécessaire puisque nous atteignons 100% de précision sans elle, mais elle rend l'explication plus convaincante.

Nous avons aussi étudié le nombre d'exemples nécessaires dans la base d'apprentissage pour obtenir de bonnes performances. Pour une base d'apprentissage comportant uniquement deux images (et 33 exemples dans la base de

test, par validation croisée imbriquée), nous obtenons au moins 99% de précision. Dans le cas extrême d'une base d'apprentissage contenant un seul exemple, les relations fréquentes seront celles dont le degré d'appartenance est plus grand que les seuils considérés (cf. section 3.2). Ainsi, tout exemple qui n'est pas une aberration devrait permettre d'obtenir de bonnes performances.

Enfin, nous observons que notre modèle obtient de meilleures performances que le réseau de neurones convolutif présenté dans [35]. Ce modèle a été entraîné sur un jeu de données plus grand que le nôtre et ne fournit pas d'explications. La méthode la plus proche de la nôtre, présentée dans [40], ne donne pas de mesure de précision comme référence. Son désavantage est qu'elle peut ne pas assigner certaines classes d'objet, ce qui arrive en moyenne une fois tous les cinq exemples. Avec notre approche, ce genre de situation est impossible car chaque variable de notre problème doit être associée à une région de l'image.

5 Conclusion et perspectives

Dans cet article, nous présentons une nouvelle méthode d'apprentissage et de raisonnement visuel dont le but est d'annoter des objets dans des images et d'expliquer les résultats obtenus. Cette tâche est formalisée comme un problème de satisfaction de contraintes floues. Cette méthode est basée sur des relations spatiales floues qui sont apprises sur un ensemble d'objets annotés et qui sont ensuite traduites en contraintes. Nous avons validé notre approche sur un jeu d'images médicales et nous avons montré qu'elle tire profit de l'apprentissage symbolique et de sa capacité à raisonner pour expliquer les résultats qu'elle produit et atteindre une précision de 99% pour une base d'apprentissage de taille 2.

Nous envisageons de développer une stratégie pour réaliser la première étape de manière plus rapide. Une première idée est de déterminer une structure hiérarchique des relations spatiales afin de pouvoir utiliser un tri topologique. Une deuxième idée est de mettre à jour l'ordre d'évaluation des relations à partir des exemples précédemment étudiés. De plus, puisque l'utilisation de la logique floue nous permet de gérer des segments imprécis, nous souhaitons utiliser une méthode de segmentation non-supervisée sur les images de tests pour initialiser les domaines de D .

Plus globalement, il s'agit d'une première étape vers une combinaison de l'apprentissage statistique et de l'apprentissage symbolique afin de créer une intelligence artificielle explicable.

Références

- [1] I. Biederman. *On the Semantics of a Glance at a Scene*. 1981.
- [2] 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.

- [3] I. Bloch. On fuzzy distances and their use in image processing under imprecision. *Pattern Recognition*, 32(11) :1873–1895, 1999.
- [4] I. Bloch. Fuzzy spatial relationships for image processing and interpretation : a review. *Image and Vision Computing*, 23(2) :89–110, 2005.
- [5] D. V. Budescu, H-H. Por, and S. B. Broomell. Effective communication of uncertainty in the ipcc reports. *Climatic Change*, 113(2) :181–200, Jul 2012.
- [6] G. C. Cawley and N. L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(Jul) :2079–2107, 2010.
- [7] M. Cayrol, H. Farreny, and H. Prade. Fuzzy pattern matching. *Kybernetes*, 11(2) :103–116, 1982.
- [8] O. Colliot, A. V. Tuzikov, R. M. Cesar, and I. Bloch. Approximate reflectional symmetries of fuzzy objects with an application in model-based object recognition. *Fuzzy Sets and Systems*, 147(1) :141 – 163, 2004.
- [9] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu. Regression forests for efficient anatomy detection and localization in ct studies. In *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, pages 106–117, 2011.
- [10] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. In *eprint arXiv :1702.08608*, 2017.
- [11] D. Dubois, H. Fargier, and H. Prade. Possibility theory in constraint satisfaction problems : Handling priority, preference and uncertainty. *Applied Intelligence*, 6(4) :287–309, 1996.
- [12] European Council. The general data protection regulation, 2016.
- [13] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4) :594–611, April 2006.
- [14] M. Garnelo and M. Shanahan. Reconciling deep learning with symbolic artificial intelligence : representing objects and relations. *Current Opinion in Behavioral Sciences*, 29 :17 – 23, 2019.
- [15] A. Gatt and E. Reiter. Simplenlg : A realisation engine for practical applications. In *Proceedings of the 12th European Workshop on Natural Language Generation*, ENLG '09, pages 90–93, 2009.
- [16] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. Explaining Explanations : An Approach to Evaluating Interpretability of Machine Learning. *ArXiv e-prints*, 2018.
- [17] D. Gunning. Explainable artificial intelligence (xai). 2017.
- [18] G. Hinton and N. Frosst. Distilling a neural network into a soft decision tree. 2017.
- [19] O. Jimenez-del Toro, H. Müller, M. Krenn, K. Gruenberg, A. A. Taha, M. Winterstein, I. Eggel, A. Foncubierta-Rodríguez, O. Goksel, A. Jakab, et al. Cloud-based evaluation of anatomical structure segmentation and landmark detection algorithms : Visceral anatomy benchmarks. *IEEE transactions on medical imaging*, 35(11) :2459–2475, 2016.
- [20] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4) :455–492, 1998.
- [21] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, pages 646–651, 2008.
- [22] C-C. Lee, P-C. Chung, and H-M. Tsai. Identifying multiple abdominal organs from ct image series using a multi-module contextual neural network and spatial fuzzy rules. *IEEE Transactions on Information Technology in Biomedicine*, 7(3) :208–217, Sep. 2003.
- [23] F-F. Li, R. VanRullen, C. Koch, and P. Perona. Rapid natural scene categorization in the near absence of attention. *Proceedings of the National Academy of Sciences*, 99(14) :9596–9601, 2002.
- [24] Z.C. Lipton. The mythos of model interpretability. *Queue*, 16(3) :30 :31–30 :57, June 2018.
- [25] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. Adiyoso Setio, F. Ciompi, M. Ghafoorian, J. A.W.M. van der Laak, B. van Ginneken, and C. I. SÁnchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42 :60 – 88, 2017.
- [26] S.M. Lundberg and S-I. Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30*, pages 4765–4774. 2017.
- [27] G. Marcus. Deep learning : A critical appraisal. *CoRR*, abs/1801.00631, 2018.
- [28] J. McClure. Goal-based explanations of actions and outcomes. *European review of social psychology*, 12(1) :201–235, 2002.
- [29] T. Miller. Explanation in artificial intelligence : Insights from the social sciences. *Artificial Intelligence*, 267 :1–38, February 2019.
- [30] S. Minton, M. D. Johnston, A. B. Philips, and P. Laird. Minimizing conflicts : a heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58(1) :161 – 205, 1992.
- [31] O. Pauly, B. Glocker, A. Criminisi, D. Mateus, A.M. Möller, S. Nekolla, and N. Navab. Fast multiple organ detection and localization in whole-body mr dixon sequences. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011*, pages 239–247, 2011.
- [32] R. Pierrard, J-P. Poli, and C. Hudelot. A fuzzy close algorithm for mining fuzzy association rules. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations*, pages 88–99, 2018.
- [33] S. J. Read and A. Marcus-Newhall. Explanatory coherence in social explanations : A parallel distributed processing account. *Journal of Personality and Social Psychology*, 65(3) :429, 1993.
- [34] M.T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you? : Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

- [35] H.R. Roth, C.T. Lee, H-C. Shin, A. Seff, L. Kim, J. Yao, L. Lu, and R.M. Summers. Anatomy-specific classification of medical images using deep convolutional nets. *arXiv preprint arXiv :1504.04003*, 2015.
- [36] H. Shin, M. R. Orton, D. J. Collins, S. J. Doran, and M. O. Leach. Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8) :1930–1943, Aug 2013.
- [37] S. Thorpe, D. Fize, and C. Marlot. Speed of processing in the human visual system. *Nature*, 381(6582) :520, 1996.
- [38] M. C. Vanegas, I. Bloch, and J. Inglada. Fuzzy constraint satisfaction problem for model-based image interpretation. *Fuzzy Sets and Systems*, 286 :1 – 29, 2016.
- [39] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata. Zero-shot Learning – a Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.
- [40] Z. Xue, S. Antani, L. R. Long, and G. R. Thoma. Automatic multi-label annotation of abdominal ct images using cbir. In *Medical Imaging 2017 : Imaging Informatics for Healthcare, Research, and Applications*, volume 10138, page 1013807, 2017.
- [41] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3) :338 – 353, 1965.

Renforcement en-ligne pour l'apprentissage conjoint de l'analyseur sémantique et du gestionnaire de dialogue d'un système d'interaction vocale

Matthieu Riou

Bassam Jabaian

Stéphane Huet

Fabrice Lefèvre

CERI-LIA, Avignon Université, France

{matthieu.riou, bassam.jabaian, stephane.huet, fabrice.lefevre}@univ-avignon.fr

Résumé

Si la conception des systèmes de dialogue a connu de nombreuses avancées ces dernières années, l'acquisition de grands ensembles de données reste une difficulté pour leur développement rapide dans le cadre d'une nouvelle tâche. L'apprentissage en-ligne est considéré dans cet article comme un moyen pratique de surmonter cette limite. Une fois les modules du système initialisés, un unique processus gère la collection des données, leur annotation et leur utilisation dans les algorithmes d'apprentissage. Il faut alors pouvoir contrôler le coût induit pour l'utilisateur lors de cet apprentissage en-ligne. Notre travail s'intéresse à l'apprentissage simultané des modules d'analyse sémantique et de gestion du dialogue. Dans ce contexte, nous proposons différentes variantes d'apprentissage conjoint qui sont testées avec des tests utilisateurs afin de confirmer que quelques centaines de dialogues d'apprentissage seulement permettent d'atteindre de bonnes performances, améliorant celles d'un système expert à base de règles. L'analyse de ces expérimentations dans l'article fait aussi apparaître des difficultés rencontrées par les entraîneurs du système pour établir une stratégie cohérente et stable durant la phase d'apprentissage.

Mots Clef

Apprentissage en-ligne, bandit contre un adversaire, apprentissage par renforcement, apprentissage sans données de référence, systèmes de dialogue.

Abstract

Design of dialogue systems has witnessed many advances lately, yet acquiring a huge dataset remains a hindrance to their fast development for a new task or language. On-line learning is pursued in this paper as a convenient way to alleviate these difficulties. After the system modules are initiated, a single process handles data collection, annotation and use in training algorithms. A new challenge is to control the cost of the on-line learning borne by the user. Our work focuses on learning the semantic parsing and dialogue management modules. In this context, we propose several variants of simultaneous learning which are tested in user trials to confirm that only a few hundred training

dialogues allow us to achieve good performance and overstep a rule-based handcrafted system. The analysis of these experiments gives us some insights, discussed in the paper, about the difficulty for the system's trainers to establish a coherent and constant behavioural strategy to enable a fast and good-quality training phase.

Keywords

On-line learning, adversarial bandit, reinforcement learning, zero-shot learning, spoken dialogue systems.

1 Introduction

Les systèmes récents de dialogue basés sur un apprentissage profond de bout en bout (*end-to-end*) offrent des résultats de recherche très prometteurs [24, 17]. Cependant, ces modèles requièrent une très grande quantité de données pour être entraînés efficacement. Il est alors difficile d'initialiser ce type de systèmes à partir de peu de données, pour un démarrage à chaud de leur développement, suivi par leur apprentissage en-ligne par des utilisateurs, tel que décrit dans [10, 13], même si certains travaux récents ont proposé des architectures de bout en bout [8, 22].

Les expérimentations menées dans cet article sont une première étape dans le développement d'un système de dialogue pour l'évaluation neuroscientifique des compétences sociales dans la communication humaine. L'objectif final est de faire interagir des utilisateurs dans une IRM (Imagerie par Résonance Magnétique) fonctionnelle avec un agent artificiel utilisant notre système, enregistré et diffusé au patient allongé. Ces expérimentations seront faites en français, sur une nouvelle tâche (voir la Section 5) pour laquelle aucune donnée n'est disponible pour le moment. Pour cette raison, la situation n'est pas la même que lorsque des corpus, librement disponibles, sont utilisables [3].

De plus, du fait de l'absence de données préalables, le système présenté repose sur une architecture classique aux capacités éprouvées pour les interactions vocales dirigées par la tâche. Différents modules en cascade permettent, à partir d'informations audio provenant de l'utilisateur, de décoder les mots à partir du signal audio (la reconnaissance automatique de la parole, ASR), puis d'en extraire une interprétation (l'analyseur sémantique, SP); cette interprétation est

combinée aux informations récoltées préalablement durant le dialogue (l'état de croyance) pour que la politique de dialogue puisse prendre une décision à propos de la prochaine action à effectuer selon certains critères (longueur du dialogue, accomplissement de la tâche, etc.). Le gestionnaire de dialogue (*dialogue management*, DM) est suivi par plusieurs opérations permettant de transmettre la réponse jusqu'à l'utilisateur : la génération en langage naturel de l'action choisie (*Natural Language Generation*, NLG) puis la synthèse vocale. L'architecture HIS (*Hidden Information State*) [25] offre un tel cadre et permet un apprentissage par renforcement de la politique du DM. Elle peut être mise en œuvre avec des algorithmes efficaces d'apprentissage en échantillons [5] et permet un apprentissage en-ligne par des interactions directes avec des utilisateurs [11]. Plus récemment, l'apprentissage en-ligne a été généralisé aux modules d'entrées/sorties (SP et NLG), avec des protocoles permettant de contrôler le coût de ces opérations durant le développement du système (comme dans [9, 12, 20, 21]). Les travaux présentés ici sont un premier effort pour combiner les apprentissages en-ligne de l'analyseur sémantique et du gestionnaire de dialogue dans une seule et même phase de développement. Le but est d'aider à accélérer et simplifier le processus d'apprentissage, mais aussi de bénéficier des améliorations conjointes des modules.

Dans les systèmes de dialogue, le SP extrait une liste d'hypothèses de concepts sémantiques à partir de la transcription d'une phrase de l'utilisateur. Cette liste s'exprime généralement sous la forme d'une séquence d'actes de dialogue (*Dialogue Act*, DA) de la forme suivante : *acte(concept=valeur)*. Les SP de l'état de l'art se basent sur des approches probabilistes et sont appris avec différentes méthodes d'apprentissage afin d'annoter des entrées utilisateurs avec ces concepts sémantiques [16, 7]. Utiliser ces techniques d'apprentissage supervisé requiert une grande quantité de données annotées qui sont dépendantes du domaine et souvent difficiles à obtenir.

Pour surmonter cet obstacle, un premier exemple de méthode proposée a été un algorithme d'apprentissage sans données de référence pour la classification de phrase sémantique [6]. Cette méthode tente de trouver un lien entre les catégories et les phrases dans un espace sémantique. Un réseau de neurones profond peut être entraîné sur une large quantité de données non annotées et non structurées afin d'apprendre cet espace sémantique. En se basant sur cette idée, [9] a présenté une méthode d'apprentissage sans données de référence pour le SP (*Zero-Shot Semantic Parser*, ZSSP) basé sur les plongements de mots [18]. Cette approche ne requiert ni donnée annotée ni donnée contextuelle et a été utilisée récemment pour différents modules de systèmes de dialogue [23, 26, 1]. En effet, le modèle est amorcé en utilisant seulement la description ontologique du domaine visé et un plongement de mots générique (appris sur un grand corpus de données ouvert librement accessible). Cependant, afin d'améliorer encore les modules avec une supervision éclairée et contrôlée des utilisateurs,

une stratégie d'apprentissage actif a été introduite en utilisant un algorithme de bandit contre un adversaire [12].

En prolongeant cette idée, un schéma d'apprentissage actif a été mis en place pour l'apprentissage du DM [11] grâce à l'algorithme par renforcement KTD [14]. Il utilise le lissage des récompenses (*reward shaping*) [19] pour prendre en compte des récompenses locales (basées sur les tours de dialogue) données par l'utilisateur pour offrir un meilleur contrôle sur le processus d'apprentissage et pour l'accélérer.

Puisque des solutions existent pour l'apprentissage actif en-ligne des deux modules SP et DM, nous considérons maintenant leur mise en application simultanée pour gérer l'apprentissage global du système. Tout d'abord, une application directe des techniques existantes est présentée et testée ; les modules et les paramètres de leur apprentissage en-ligne restent séparés (un algorithme de bandit pour le SP, un *Q-learner* pour le DM). Puis une nouvelle proposition avec des paramètres partagés dans un seul *Q-learner* est aussi introduite et évaluée.

Dans la suite de cet article complétant notre précédente publication [21], nous présentons les méthodes d'apprentissage en-ligne du SP dans la Section 2 et du DM dans la Section 3. Nous définissons ensuite les stratégies d'apprentissage en-ligne conjoint dans la Section 4. La Section 5 présente une étude expérimentale avec des évaluations humaines des approches proposées, accompagnées par une analyse nous fournissant des informations pertinentes sur les implications concrètes de l'apprentissage en-ligne. Nous concluons dans la Section 6.

2 Apprentissage en-ligne du Zero-Shot SP

Le modèle SP utilisé dans cet article est le modèle ZSSP décrit dans [12]. Le modèle utilise une base de connaissances sémantiques K et un espace de représentation sémantique F . K contient certains exemples de blocs lexicaux associés chacun à un acte de dialogue DA, tandis que F est une représentation issue d'un plongement de mots appris avec des réseaux de neurones sur une grande quantité de données non annotées d'un domaine ouvert [18, 2]. Le modèle ZSSP construit un graphe pondéré des hypothèses à partir des phrases de l'utilisateur. Un algorithme de recherche du meilleur chemin est utilisé pour trouver la meilleure hypothèse d'annotation sémantique pour la phrase utilisateur considérée.

Le modèle est appris avec une stratégie d'adaptation en-ligne, facilitée par l'approche sans données de référence (*zero-shot*), comme présenté dans [12] et brièvement résumé ici. À chaque itération du dialogue, le système choisit une action d'adaptation $i_t \in \mathcal{I}$ et utilise les retours de l'utilisateur pour mettre à jour K . Les gains du système $g(i_t)$, l'effort de l'utilisateur $\phi(i_t)$ et la fonction de coût $l(i_t)$ sont définis ci-dessous et peuvent être estimés durant l'apprentissage en-ligne.

Trois actions sont possibles :

- **Skip** : Ignore la phase d'adaptation pour ce tour ($\phi(\text{skip}) = 0$).
- **AskConfirm** : Une question oui/non est posée à l'utilisateur à propos de la validité des DA sélectionnés par la meilleure hypothèse sémantique. Si toute la phrase est validée, $\phi(\text{YesNoQuestions}) = 1$. Sinon, $\phi(\text{YesNoQuestions})$ est égal à 1+ le nombre de DA dans la meilleure hypothèse sémantique (une demande de confirmation oui/non par DA).
- **AskAnnotation** : l'utilisateur est invité à réannoter la phrase entière.
 $\phi(\text{AskAnnotation}) = 1$ si la phrase est validée dans son ensemble. Sinon, l'utilisateur devra d'abord spécifier les blocs qu'il souhaite annoter (+1 par frontière spécifiée), et ensuite le système demandera, itérativement pour chaque DA, les actes, concepts et valeurs si nécessaire (+1 pour chaque question intermédiaire).

Afin de minimiser le nombre de demandes d'annotation et ainsi l'effort demandé à l'utilisateur, un algorithme de bandit contre un adversaire est utilisé pour trouver i_1, i_2, \dots, i_t tels que pour chaque t , le système minimise le coût $l(i_t)$. La fonction de coût $l(i) \in [0, 1]$ est calculée comme suit :

$$l(i) := \underbrace{\gamma g(i)}_{\text{amélioration du système}} + (1 - \gamma) \underbrace{\frac{\phi(i)}{\phi_{max}}}_{\text{effort de l'utilisateur}}$$

où $\gamma \in [0, 1]$ équilibre l'importance de l'amélioration du système avec celle de l'effort de l'utilisateur pour le système, et $\phi_{max} \in \mathbb{N}^*$ est le nombre maximum d'échanges entre le système et l'utilisateur (dans un même tour de dialogue). Dans ce travail, la valeur de γ a été fixée à 0.5.

3 Apprentissage en-ligne du gestionnaire de dialogue par renforcement

Le gestionnaire de dialogue utilisé dans cet article adapte un système présenté dans [11]. Il utilise un *framework* de gestion de dialogue basé sur un POMDP (*Partially Observable Markov Decision Process*), le *Hidden Information State* (HIS) [25]. Dans ce *framework*, le système tient à jour une distribution sur les états possibles du dialogue (l'état de croyance) et l'utilise pour générer une réponse appropriée. Un algorithme d'apprentissage par renforcement (*reinforcement learning*, RL) est utilisé pour entraîner le système en maximisant une récompense cumulée pondérée.

À chaque tour, le gestionnaire de dialogue génère plusieurs réponses possibles selon son état de croyance. En pratique il génère toujours 11 actes de dialogue, correspondant aux 11 actes résumés suivants : Greet, Bye, Bold Request, Tentative Request, Confirm, Find Alternative, Split, Repeat, Offer, Inform and Request. De plus, certains actes peuvent être étiquetés comme impossibles à certains moments si aucune conversion vers une action com-

plète n'est possible (par exemple, Inform dans le cas où aucune entité n'est encore sélectionnée).

Le gestionnaire de dialogue choisit ensuite le meilleur acte résumé pour le contexte donné. Pour apprendre cette politique, une approche d'apprentissage par renforcement est utilisée : l'algorithme d'apprentissage KTDQ [15], dérivée du cadre des *Kalman-based Temporal Differences* (KTD). À chaque tour, la politique sélectionne un acte résumé pour répondre à l'utilisateur, puis ce dernier donne un retour permettant d'attribuer un score à la réponse et de mettre à jour la politique. Deux types de retours sont possibles. Le retour global est donné à la fin du dialogue en demandant à l'utilisateur si le dialogue dans son ensemble est un succès ou non. Le retour direct utilisateur s_i est donné à chaque tour i pour attribuer un score seulement à la dernière réponse. Il est composé de deux parties : le score donné par l'utilisateur à cette dernière réponse (appelé retour additionnel a_i) moins la fonction Ψ (qui prend en compte l'historique d'annotation pour lisser ce retour local), et le coût du tour qui permet de pénaliser les dialogues trop longs en ajoutant un score négatif à chaque tour (appelé retour f_i) : $s_i = f_i + (\theta a_i - \Psi)$.

Ici Ψ est le retour additionnel du tour précédent a_{i-1} et $\theta = 0.95$. À la fin du dialogue, la politique est mise à jour selon la totalité des retours collectés. Dans ce travail, la valeur du retour global est fixée à 20 en cas de succès et 0 sinon. Le retour f_i est fixé à -1 pour chaque tour et le retour additionnel $a_i \in \{-1, -0.5, 0, 0.5, 1\}$.

4 Apprentissage en-ligne conjoint

Afin d'apprendre en-ligne efficacement, l'utilisateur doit être capable d'améliorer à la fois l'analyseur sémantique et le gestionnaire de dialogue. Dans ce but, deux protocoles d'apprentissage différents sont proposés.

Le premier, appelé **BR**, juxtapose le bandit permettant d'apprendre le ZSSP et l'approche *Q-learner* RL permettant d'apprendre le gestionnaire de dialogue¹. Un algorithme de bandit contre un adversaire tel que décrit dans la Section 2 est appliqué pour l'apprentissage du ZSSP, et un *Q-learner* est utilisé pour apprendre la politique du DM, tel que décrit dans la Section 3.

Le second protocole, appelé **RR**, insère directement les actions d'apprentissage du ZSSP dans la politique RL du gestionnaire de dialogue. Il combine ainsi les deux processus d'apprentissage dans une seule politique².

Cette variante de l'apprentissage conjoint fusionne les deux politiques dans un seul *Q-learner*. Dans ce but, le vecteur d'états résumés du DM est augmenté avec des dimensions liées au ZSSP. Pour limiter cet agrandissement, une seule dimension est ajoutée. Cette nouvelle dimension est calculée à partir d'un ensemble d'indices de qualité des annotations faites par le modèle ZSSP. Sur une échelle de 3 points, cinq caractéristiques sont utilisées :

1. BR correspond au protocole Bandit-SP et RL-DM.

2. RR correspond à un système appris avec un RL-SP et un RL-DM.

1. **confidence** : le score de confiance de l'analyseur sémantique sur $[0, 1]$.
2. **fertility** : le ratio du nombre de concepts sur le nombre de mots dans la phrase sur $[0, 1]$, puisque le ZSSP tend à produire une sur-segmentation en insérant des concepts.
3. **rare** : la présence ou non (binaire) de concepts rares dans l'annotation. Ces concepts rares sont *help*, *repeat*, *restart*, *reqalts*, *reqmore*, *ack* or *thankyou*, et sont souvent mal annotés.
4. **known chunks** : le ratio du nombre de blocs annotés présents dans la base de connaissances sémantiques K sur le nombre total de blocs annotés, compris dans $[0, 1]$.
5. **gap** : la différence entre les scores de confiance de la première et de la deuxième annotations. Puisque ces différences sont très faibles (< 0.01), un logarithme naturel est appliqué pour éclater les données.

La dimension liée au ZSSP est calculée à partir de ces caractéristiques comme suit :

- 0 *all clear* : rare = 0 et confidence ≤ 0.499 et fertility ≤ 0.4 et known chunks ≥ 0.5 et gap ≥ -5.5
- 1 *average condition* : rare = 0 et fertility ≤ 0.5 et known chunks ≥ 0.15 et gap ≥ -6.5 et (confidence > 0.499 ou fertility > 0.4 ou known chunks < 0.5 ou gap < -5.5)
- 2 *alarming* : rare = 1 ou fertility > 0.5 ou known chunks < 0.15 ou gap < -6.5

Dans le protocole RR, les deux actions d'annotation du ZSSP (*Askconfirm* et *Askannotation*, voir Section 2) sont aussi incluses dans la liste des actions résumées qui peuvent être sélectionnées par la politique de dialogue. Dans ce cas, la fenêtre appropriée d'annotation est présentée à l'utilisateur dans l'interface graphique du système et permet de corriger l'annotation proposée. L'utilisation d'interactions purement vocales pour ce processus est à l'étude. Même si c'est réalisable, cela reste une tâche compliquée qui pourrait introduire de nouvelles erreurs ; ainsi il semble plus approprié d'évaluer dans un premier temps le processus en utilisant une interface graphique limitant les erreurs. Une fois l'annotation faite, le tour est mis à jour (l'annotation a donc pris la place d'une réponse orale normale de l'utilisateur) et le dialogue reprend. Même si la politique serait capable de l'apprendre d'elle-même, nous avons choisi de désactiver les actions d'annotation en séquence (elles sont étiquetées comme impossibles au tour suivant). Finalement, ces deux actions d'annotation du ZSSP ont un retour social spécifique : au lieu d'être fixé à -1 , le retour f_i utilise la fonction de coût $l(i)$ définie dans la Section 2 et réajustée pour obtenir un score $\in [-1, 1]$: $f_i = (1.0 - l_i) \times 2 - 1$.

5 Étude expérimentale

5.1 Description de la tâche

Les expériences présentées dans cet article concernent un système de tchat reformulé pour traiter une tâche orientée vers un but. Dans ce contexte, les utilisateurs discutent avec le système à propos d'une image (sélectionnée parmi un petit ensemble prédéfini de 6 images), et essaient de deviner le message porté par l'image en coopération avec le système, comme décrit dans [4]. Afin d'utiliser un système orienté par la tâche, l'application du système a été construite autour d'une base de données contenant plusieurs centaines de combinaisons possibles des caractéristiques de l'image. Chaque combinaison est associée à une hypothèse sur le message envoyé. Durant l'interaction avec le système, il est attendu que l'utilisateur apporte progressivement des éléments de l'image correspondant à une entité dans la base de données. De ce fait, le système sélectionne un petit sous-ensemble des entités possibles depuis lesquelles il peut proposer des caractéristiques additionnelles pour informer l'utilisateur et, ultimement, un message prédéfini proposant une explication plausible de l'objectif de l'image. Cela permet à l'utilisateur de parler assez librement de l'image pendant quelques dizaines de secondes, avant d'argumenter brièvement sur le message. Aucune argumentation n'est possible du côté du système, celui-ci peut seulement proposer un message prédéfini et la discussion est censée durer autour d'une minute au plus.

La base de données, dépendante de la tâche et utilisée dans cette expérience, est dérivée de la tâche de description d'images de fruits proposée par l'INT [4], ainsi que des tâches génériques de dialogue. La sémantique du domaine est décrite par 16 types d'actes, 9 concepts et 51 valeurs. Les 53 formes lexicales utilisées pour modéliser les actes ont été élaborées manuellement.

5.2 Résultats

Cette section présente l'évaluation des deux approches d'apprentissage conjoint. Deux systèmes complémentaires sont proposés en comparaison : **ZH** est un système *baseline* sans apprentissage en-ligne qui utilise le ZSSP initial et une politique de gestionnaire de dialogue prédéfinie par un expert, tandis que **BH** combine l'apprentissage en-ligne par bandit du ZSSP avec la politique prédéfinie du gestionnaire de dialogue.

Trois experts différents ont appris un modèle de chaque approche. Pour chaque apprentissage, l'utilisateur expert communique avec le système pour entraîner le modèle. Puis un groupe de 11 utilisateurs naïfs (ainsi que 2 utilisateurs experts supplémentaires pour le modèle **ZH**) ont testé chaque modèle. À la fin de chaque session, les utilisateurs devaient indiquer si le dialogue était un succès ou non, et donner un score entre 0 (le pire) à 5 (le meilleur) aux capacités de compréhension et de génération du système. La quantité de dialogues pour l'apprentissage et le nombre de tests pour chaque configuration sont donnés dans le Ta-

Modèle	Apprentissage (#dial)	Test (#dial)	Succès (%)	Récompense moy. cum.	Score de compréhension du système	Score de génération du système
ZH	0	142	29	-1,9	1,6	4,0
BH	80	96	70	7,0	3,2	4,6
BR	140	96	89	10,9	3,3	4,6
RR	140	96	65	4,4	2,9	3,8

TABLE 1 – Évaluation des différentes configurations de l'apprentissage en-ligne

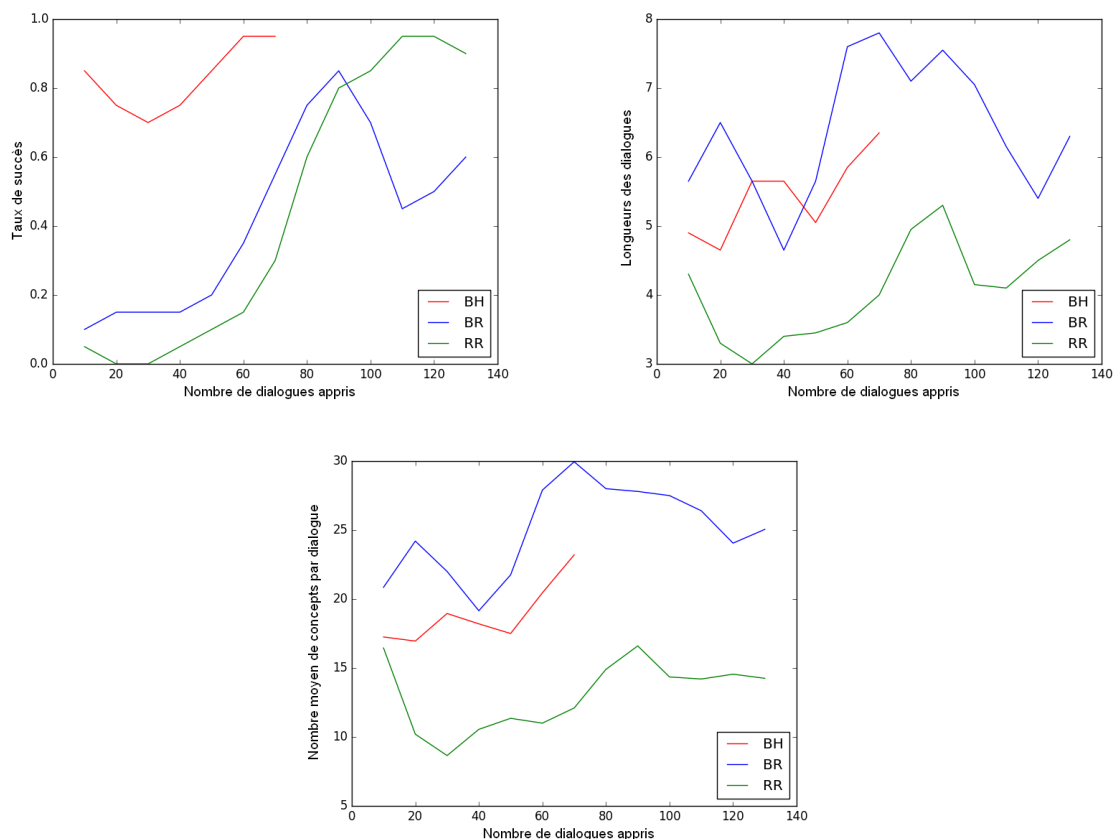


FIGURE 1 – (a) Taux de succès (b) Longueurs des dialogues (c) Nombres de concepts par dialogue durant la phase d'apprentissage pour les différents modèles

bleau 1, aux colonnes 2 et 3.

Les résultats des tests utilisateurs pour les deux apprentissages de chaque approche sont présentés dans le Tableau 1. Ces résultats montrent que les différentes configurations du système offrent des performances acceptables. Le modèle BR, entraîné sur 140 dialogues, montre le meilleur taux de succès (89%), significativement³ meilleur que les autres modèles. De plus, le modèle ZH donne des taux de succès significativement³ moins bons que les autres modèles. La différence de performances entre les modèles ZH et BH (+41 points) montre un impact de l'apprentissage du ZSSP sur le succès global d'une conversation, visible aussi dans

3. La significativité statistique a été calculée avec un test t de Welch bilatéral. Les résultats sont considérés significatifs pour une p-valeur < 0,001.

le score de compréhension (1,6 pour ZH contre 3,2 pour BH). La récompense cumulée moyenne dans les tests est directement corrélée avec le taux de succès et confirme les conclusions précédentes. De plus, en raison de l'ajustement du système de génération à base de patrons, le score de génération du système est élevé ($\geq 3,8$) pour toutes les configurations.

Le protocole RR donne un taux de succès inférieur à BH et BR (65% pour RR contre 89% pour BR). En étudiant l'historique de l'apprentissage, il semble que cela soit dû à un seuil de déclenchement des actions d'apprentissage du ZSSP dans RR beaucoup trop bas après la période d'exploration par rapport à l'utilisation du bandit dans BH et BR. Pour y remédier, l'état de la politique doit être modifié pour mieux prendre en compte les situations qui devraient mener

à des actions d’annotations, tout en préservant ses capacités de discriminations pour les actes de dialogue. Mais cette approche reste à développer et à améliorer, puisqu’elle permet de simplifier grandement le développement de système en étant basée sur un unique *framework* pour l’apprentissage conjoint.

5.3 Analyse de l’apprentissage en-ligne

Afin d’améliorer le protocole d’apprentissage et d’extraire des informations pertinentes de nos expériences, les historiques des apprentissages ont été analysés. Ces apprentissages nous permettent d’avoir une meilleure idée des propriétés du processus d’apprentissage en-ligne et des stratégies implémentées par les experts. Ces derniers étaient assez libres dans leur choix d’apprentissage. Par exemple, ils n’ont eu aucune restriction sur la façon d’utiliser les retours additionnels.

En ce qui concerne la politique du DM, tous les experts disent avoir préféré utiliser dans un premier temps des dialogues simples pour construire une politique efficace. Puis, lorsque le système a commencé à être utilisable (des séquences régulières de 2-3 dialogues ayant été réussies), ils ont proposé des dialogues plus sophistiqués pour étendre les capacités du système. Ces choix sont visibles dans les historiques des apprentissages. Dans les figures 1 (a) et (b), on peut observer que le taux de succès et la longueur des dialogues varient énormément durant l’apprentissage⁴ : au début, les dialogues sont courts (avec un minimum de 3 tours pour RR et autour de 6 tours pour BH et BR) et le taux de succès augmente progressivement. Puis, après 40 à 60 dialogues, ils deviennent plus complexes, menant à une diminution de la récompense et du taux de succès, ainsi qu’à de plus longs dialogues. Notons aussi que pour le modèle BH, les experts produisent plus rapidement des dialogues sophistiqués puisque ce modèle ne requiert pas d’apprendre la politique du DM.

Pour approfondir l’analyse de l’apprentissage du ZSSP, nous considérons le nombre total de concepts discutés par dialogue durant l’apprentissage. Pour compter les concepts discutés, chaque triplet acte/concept/valeur a été considéré comme un concept. Les résultats sont présentés dans la Figure 1 (c). Durant les premières étapes, lorsque les experts utilisent des dialogues simples, ils ont plutôt tendance à limiter le nombre de concepts échangés. Puis, au fur et à mesure de l’amélioration du système, ils diversifient les concepts utilisés. L’approche RR montre moins de concepts par dialogue que BH et BR. Cela peut être expliqué par le seuil de déclenchement des actions d’apprentissage du ZSSP très bas pour ce modèle ; ainsi les experts n’ont pas pu étendre efficacement les concepts utilisés durant les dialogues, et ont eu tendance à conserver des expressions en évitant d’ajouter trop de bruits dans les modèles.

Les retours additionnels permettent aux experts de récompenser localement ou pénaliser des réponses spécifiques du

système, en complément de la récompense globale (conditionnée au succès de la tâche et pénalisée par la longueur du dialogue). [11] a montré que dans un environnement simulé, les retours négatifs permettent de mieux guider l’apprentissage du dialogue que les retours positifs et améliore la convergence du processus d’apprentissage. Or, nos expériences révèlent que les experts ont instinctivement tendance à plus souvent pénaliser les mauvaises réponses que récompenser les bonnes. On peut l’observer dans la Figure 2 : les retours négatifs sont utilisés régulièrement pour BR et RR (la politique de BH étant prédéfinie, elle n’implique pas de retours utilisateurs). À l’inverse, l’utilisation des retours positifs est plus variable : presque aucun pour BR et plus de retours positifs que négatifs pour RR. Même si l’utilisation élevée de retours positifs demande plus d’efforts aux experts, cela permet aussi d’améliorer la stabilité des choix valides dans la politique de dialogue.

Afin de trouver des améliorations possibles pour le processus d’apprentissage, nous avons étudié les causes régulières d’échecs des dialogues. 48 dialogues de test ont été analysés en détail. Une grande différence des taux de succès entre les modèles peut être expliquée par des erreurs d’annotation de l’analyseur sémantique. Pour étudier cet aspect, une annotation des sorties correctes de l’analyseur sémantique dans le sous-ensemble de dialogues de test a été réalisée par un expert, en précisant en cas d’erreur si elle était due à une erreur de reconnaissance vocale (empêchant toute interprétation correcte ensuite). Ainsi, un score de succès du SP a été calculé comme le ratio du nombre de concepts correctement produits sur le nombre de concepts dans la référence plus le nombre de concepts insérés. ZH présente un taux de succès du SP de 41% contre 70% pour BH, 60% pour BR et 49% pour RR. Ces variations permettent d’expliquer les différences des taux de succès entre les différents modèles.

En regardant des exemples spécifiques d’erreurs du SP (voir Tableau 2), on observe que certaines erreurs sont régulièrement responsables d’échecs de dialogues, notamment :

- une fausse identification d’un acte au *revoir*, souvent due à des erreurs de reconnaissance vocale, entraînant un arrêt prématuré du dialogue ;
- des erreurs du SP créant de fausses croyances dans le gestionnaire de dialogue, poussant le système à croire que l’utilisateur parle d’une autre image. Ces erreurs sont parfois dues à des erreurs de reconnaissance vocale : environ 10% pour ZH, 25% pour BH, 28% pour BR et 27% pour RR. Mais ces valeurs doivent être liées aux taux de succès du SP : un meilleur SP implique moins d’erreurs et donc une plus grande proportion d’erreurs causées directement par la reconnaissance vocale.

De plus, les analyseurs sémantiques proposent toujours des résultats variables pour les phrases négatives. Par exemple « ce n’est pas un citron » n’est pas comprise par les modèles ZH et RR, mais l’est généralement dans BH et BR.

4. Ces taux sont calculés sur une fenêtre glissante de 20 dialogues.

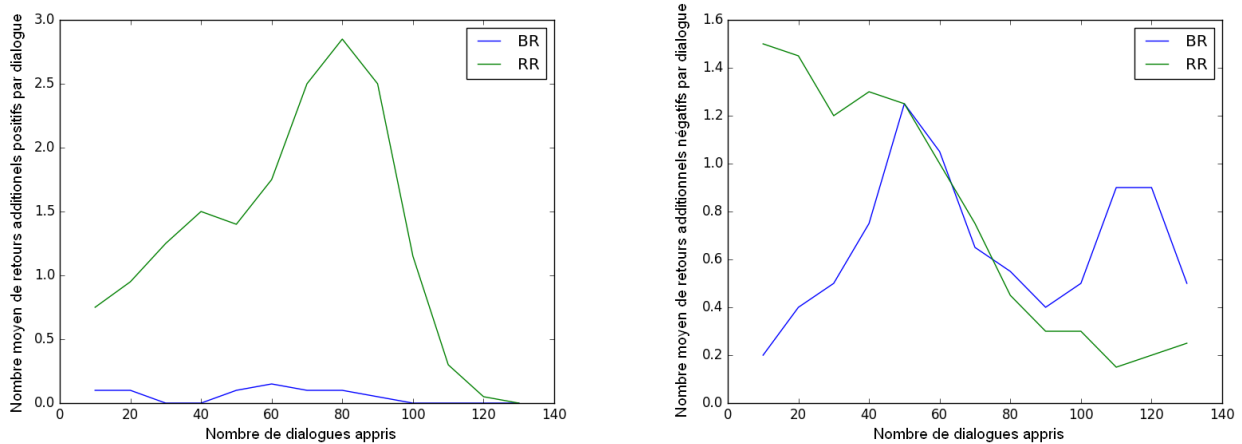


FIGURE 2 – (a) Nombre moyen de retours additionnels positifs par dialogue (b) Nombre moyen de retours additionnels négatifs par dialogue durant l'apprentissage pour les différents modèles

TABLE 2 – Exemples d'erreurs du SP impactant les dialogues.

ASR	il y a des bras des dieux et des jambes <i>L'erreur ASR est due à la proximité phonétique entre dieux et yeux .</i>
SP	inform(looks_like=superhero,possesses=legs,possesses=arms) <i>Le SP associe incorrectement dieux au concept de super-héros.</i>
ASR	le fruit a l'air d'avoir baissé les bras
SP	inform(possesses=bras), requalts() <i>Les expressions figées sont difficiles à interpréter, entraînant des erreurs du SP. De plus, le SP a parfois tendance à ajouter des actes comme requalts() ou ack() (en particulier pour ZH et RR).</i>

Néanmoins, « ce n'est pas un citron, c'est une pomme » reste toujours mal gérée, quel que soit le modèle.

6 Conclusion

Après avoir proposé des méthodes pour entraîner de manière interactive en-ligne à la fois l'analyseur sémantique et le gestionnaire de dialogue, cet article a proposé et évalué différentes manières de combiner ces méthodes dans un processus d'apprentissage conjoint. Les expérimentations ont été menées en conditions réelles et sont donc peu nombreuses. Toutefois, elles ont permis de montrer qu'un apprentissage conjoint peut être effectué, et qu'après environ une centaine de dialogues, les performances des diverses configurations testées étaient généralement satisfaisantes par rapport à un système expert. En outre, ces expérimentations ont permis de relever des informations pertinentes sur les caractéristiques du processus d'apprentissage en-ligne. Par exemple, dans nos expérimentations les experts se concentrent plus souvent sur des dialogues simples dans un premier temps pour augmenter le taux de succès, puis essaient ensuite de produire des dialogues plus complexes pour améliorer à la fois le gestionnaire de dialogue et l'analyseur sémantique. De plus, les retours négatifs sont généralement préférés afin de guider le processus d'apprentis-

sage, ce qui est un bon point puisqu'une précédente étude a montré en simulation qu'ils étaient plus profitables pour le processus d'apprentissage [11].

En se basant sur ces résultats, une prochaine tâche consistera à proposer une fusion des politiques issues de différents apprentissages, afin d'être en mesure de construire des données d'apprentissage issues de différents utilisateurs et d'économiser ainsi encore plus de temps aux développeurs du système.

Remerciements

Ce travail a été partiellement soutenu par un financement de l'ANR-16-CONV-0002 (ILCB) et ANR-11-LABX-0036 (BLRI).

Références

- [1] A. Bapna, G. Tur, D. Hakkani-Tur, and L. Heck. Towards zero-shot frame semantic parsing for domain scaling. *arXiv preprint arXiv :1707.02363*, 2017.
- [2] J. Bian, B. Gao, and T. Liu. Knowledge-powered deep learning for word embedding. In *ECML*, 2014.
- [3] I. Casanueva, P. Budzianowski, P.-H. Su, N. Mrkšić, T.-H. Wen, S. Ultes, L. Rojas-Barahona, S. Young, and M. Gašić. A Benchmarking Environment for

- Reinforcement Learning Based Task Oriented Dialogue Management. *arxiv.org*, nov 2017.
- [4] T. Chaminade. An experimental approach to study the physiology of natural social interactions. *Interaction Studies*, 18(2) :254–276, 2017.
- [5] L. Daubigney, M. Geist, S. Chandramohan, and O. Pietquin. A comprehensive reinforcement learning framework for dialogue management optimization. *Selected Topics in Signal Processing*, 6(8) :891–902, 2012.
- [6] Y. Dauphin, G. Tur, D. Hakkani-Tur, and L. Heck. Zero-shot learning and clustering for semantic utterance classification. *arXiv preprint arXiv :1401.0509*, 2014.
- [7] A. Deoras and R. Sarikaya. Deep belief network based semantic taggers for spoken language understanding. In *INTERSPEECH*, 2013.
- [8] B. Dhingra, L. Li, X. Li, J. Gao, Y.-N. Chen, F. Ahmed, and L. Deng. Towards end-to-end reinforcement learning of dialogue agents for information access. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, volume 1, pages 484–495, 2017.
- [9] E. Ferreira, B. Jabaian, and F. Lefèvre. Online adaptive zero-shot learning spoken language understanding using word-embedding. In *ICASSP*, 2015.
- [10] E. Ferreira and F. Lefèvre. Expert-based reward shaping and exploration scheme for boosting policy learning of dialogue management. In *ASRU*, 2013.
- [11] E. Ferreira and F. Lefèvre. Reinforcement-learning based dialogue system for human-robot interactions with socially-inspired rewards. *Computer Speech & Language*, 34(1) :256–274, 2015.
- [12] E. Ferreira, A. Reiffers-Masson, B. Jabaian, and F. Lefèvre. Adversarial bandit for online interactive active learning of zero-shot spoken language understanding. In *Proceedings of ICASSP*, 2016.
- [13] M. Gašić, C. Breslin, M. Henderson, D. Kim, M. Szummer, B. Thomson, P. Tsiakoulis, and S. Young. On-line policy optimisation of bayesian spoken dialogue systems via human interaction. In *IEEE ICASSP*, pages 8367–8371, 2013.
- [14] M. Geist and O. Pietquin. Kalman temporal differences. *Artificial Intelligence Research*, 39(1) :483–532, Sept. 2010.
- [15] M. Geist and O. Pietquin. Managing uncertainty within value function approximation in reinforcement learning. In *Active Learning and Experimental Design workshop (collocated with AISTATS 2010)*, *Sardinia, Italy*, volume 92, 2010.
- [16] S. Hahn, M. Dinarelli, C. Raymond, F. Lefèvre, P. Lehnen, R. De Mori, A. Moschitti, H. Ney, and G. Riccardi. Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE TASLP*, 19(6) :1569–1583, 2010.
- [17] X. Li, Y.-N. Chen, L. Li, J. Gao, and A. Celikyilmaz. End-to-end task-completion neural dialogue systems. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, volume 1, pages 733–743, 2017.
- [18] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, 2013.
- [19] A. Ng, D. Harada, and S. Russell. Policy invariance under reward transformations : Theory and application to reward shaping. In *ICML*, 1999.
- [20] M. Riou, B. Jabaian, S. Huet, and F. Lefèvre. Online adaptation of an attention-based neural network for natural language generation. In *Proceedings of INTERSPEECH*, 2017.
- [21] M. Riou, B. Jabaian, S. Huet, and F. Lefèvre. Joint On-line Learning of a Zero-shot Spoken Semantic Parser and a Reinforcement Learning Dialogue Manager. In *IEEE ICASSP*, 2019.
- [22] P. Shah, D. Hakkani-Tur, B. Liu, and G. Tur. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 3 (Industry Papers)*, volume 3, pages 41–51, 2018.
- [23] S. Upadhyay, M. Faruqui, G. Tür, H.-T. Dilek, and L. Heck. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6034–6038. IEEE, 2018.
- [24] T.-H. Wen, D. Vandyke, N. Mrkšić, M. Gasic, L. M. Rojas Barahona, P.-H. Su, S. Ultes, and S. Young. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers*, pages 438–449, 2017.
- [25] S. Young, M. Gašić, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, and K. Yu. The hidden information state model : A practical framework for pomdp-based spoken dialogue management. *Computer Speech and Language*, 24(2) :150–174, 2010.
- [26] T. Zhao and M. Eskenazi. Zero-shot dialog generation with cross-domain latent actions. *arXiv preprint arXiv :1805.04803*, 2018.

Insights about user-centric contextual online adaptation of coordinated multi-agent systems in smart homes

Iago Felipe Trentin^{1,2}

Olivier Boissier²

Fano Ramparany¹

¹ Orange Labs, Meylan, France 38240

² Univ. Lyon, MINES Saint-Étienne, CNRS Lab Hubert Curien UMR 5516, Saint-Étienne, France F-42023

iagofelipe.trentin@orange.com / iago.felipe.trentin@gmail.com

Résumé

Avec le développement des maisons intelligentes, il est de plus en plus nécessaire de déployer des systèmes capables de s'adapter à des contextes fortement dynamiques ainsi qu'aux besoins des utilisateurs. Ces maisons sont de plus en plus équipées d'objets connectés qui manquent souvent de l'intelligence nécessaire à leur adaptation et coordination en ligne. Cet article a pour objectif d'analyser des approches susceptibles de répondre à ces défis et d'identifier des directions possibles de travaux futurs pour développer des systèmes multi-agents capables de combiner planification et action, en étant sensibles au contexte et dirigés par des buts, et qui pourront inspirer les générations futures de maisons intelligentes.

Mots Clef

Systemes multi-agent, Planification, Action, Coordination, Adaptation, Maison intelligente

Abstract

Smart homes are becoming a reality, and with that the need to deploy systems capable of adapting to highly dynamic contexts in order to satisfy user needs. These homes are increasingly permeated by internet of things objects, but often they are not embedded with the artificial intelligence necessary for their adaptation and coordination at run time. This paper aims to present promising approaches for these challenges, and identify possible directions for future work in developing a coordinated responsible goal-driven context-aware planning and acting multi-agent system, which may inspire future generations of smart homes.

Keywords

Multi-agent systems, Planning, Acting, Coordination, Adaptation, Smart home

1 Introduction

With the advance of ubiquitous technologies, the physical objects and environments of our daily life are becoming more intelligent, capable of perceiving and acting in

the world. But for the time being, such technologies have difficulties adapting to highly dynamic environments such as smart homes, and users also struggle to make different objects work together. This paper is motivated by a current smart home challenge: the ability to adapt to context in order to satisfy user evolving needs. Such complex behavior can only be achieved by embedding Artificial Intelligence (AI) techniques into Internet of Things (IoT) objects.

To tackle the aforementioned challenge, there are solutions in the literature that allow embedding intelligent entities (a.k.a. agents) with: acting capabilities, i.e. act in the world and react to predicted changes; and also planning capabilities, i.e. reason upon current context to find new solutions to a given need. In a smart home, for maximum user satisfaction, agents should be able to timely adapt to user needs, with minimum conscious and direct input from occupants. This online adaptation may be possible by endowing agents with planning and acting functions designed to execute together at run-time.

In this paper, we aim to bring up and renew questionings about adaptation capabilities of IoT objects, specifically focusing on online adaptation of plans and actions in a user-centric manner that respects user goals and user values. We analyze current state-of-the-art solutions, organizing them according to common properties we find in the literature. The understanding and exploring of the state-of-the-art approaches presented may allow future work to create a decentralised and coordinated planning and acting system. It would be: *user-centric*, gravitating towards enhancing human experience; *goal-driven*, intelligently reasoning about and pursuing user objectives; *data-driven*, adapting at run-time to environmental and user inputs; and *trustworthy*, being consistent with user values and expectations.

Based on these challenges we analyze the state of the art as follows: Section 2 presents the challenges we are interested in, with special focus on the smart home domain; Section 3 presents well-established solutions, especially in the planning and acting domains, that have potential to help addressing our challenges; Section 4 shows the interest in uniting planning and acting to ultimately have a synergistic

solution; Section 5 identifies complementary domains, the goal and value reasoning, that add valuation to such a solution; Section 6 expands the envisioned solution to consider interaction and coordination in a decentralised system.

2 Motivation

This section elaborates the above stated challenges and interest in developing a system capable of adapting itself to context and users, in IoT environments such as smart homes. We consider publicly available solutions, their current state and limitations, that partially fulfill identified smart home requirements. It also shows aspects that can be improved or redesigned to better address our requirements. Since it explores a large domain, this paper is not expected to be an exhaustive study of the available literature.

2.1 Current IoT solutions

Current smart homes are becoming increasingly more complex, permeated by ubiquitous and heterogeneous elements designed by different vendors [1]. Some solutions help with discovery of objects and composition of new services mixing available ones [1, 2], other popular solutions allow the development of IoT mashups¹. For a great smart home user experience, the composition and delivery of user-centric services are important.

Although there is interest in ad-hoc mashups, often temporary and non-scalable, for small tactical and specific needs [3], such applications do not present reasoning nor adaptive behavior. They simply perform tasks previously programmed by humans. Also, with discovery and composition of services but no intelligent entities to demand and use them, user experience tends to become cumbersome – due to constant need for manual actions, like system re-configuration, user input, etc.

An IoT sub-domain, the growing Web of Things (WoT), allows objects to connect by means of the existing Web infrastructure [4]. Nevertheless, the resulting system has a priori no coordinating abilities, therefore provided services may not be optimal, or can even conflict with each other.

2.2 Intelligent agents

In the previous section, we discussed the benefits in embedding intelligence into objects, giving them adaptive, reasoning, coordinating and self-managing capacities, which can be done by the use of AI techniques. In the AI field, an intelligent agent is a goal-driven entity capable of perceiving and acting upon its environment [5]. An agent must have acting capabilities in order to interact with its environment, and this connection may have many forms.

An agent with direct and tight coupling to its environment performs reactive acting, i.e. a data-driven behavior where actions follow predefined models and plans.

One with a more flexible coupling performs adaptive acting, i.e. contextual changes trigger refinement² of avail-

able actions to find possible alternatives to continue current plans, individually or collectively with other agents [6, 7].

Going further, an agent with a freely and relaxed coupling has a proactive behavior, presenting planning and reasoning capabilities. It is then able to reason upon available context information in order to find new solutions to a given need, allowing it to create new plans to achieve goals, which improves its adaptability.

Numerous solutions for industrial scenarios, such as manufacturing lines [8], use intelligent agents, but these environments are predictable by design, being composed of agents that perform regular, repetitive and controlled processes. In highly dynamic scenarios, like smart homes, reasoners need to be both timely and context-aware in order to answer user needs fast and autonomously.

For an agent to be context-aware, it needs to perceive and understand the world. In order to do so, context collection, modeling and reasoning techniques [9] are needed, specially in the IoT domain where sensors generate enormous amounts of raw data [10].

We are interested in dynamic context modeling techniques because they present open and expandable models that can fit wide ranges of IoT environments [11], and we know smart homes are highly heterogeneous and dynamic ones. In this paper we will consider that context is given ready-to-use by an external function.

Based on the analysis of this section, the agent of interest in a smart home is one capable of adaptive acting and context-aware planning.

2.3 Challenges and requirements

An advanced smart home is expected to perform responsive contextual adaptation in a user-centric manner. The use of agents to tackle this challenge allows us to identify some smart home requirements.

As said previously, such environments are highly dynamic, and the system's main objective is to satisfy user needs. Its final architecture must also take into account heterogeneous objects, while allowing flexible interaction among autonomous agents, creating a social IoT ecosystem [12], which we can call a *multi-agent based smart home*.

To identify smart home requirements the following scenario is presented: a user watching a movie receives a call.

In this case, the smart home has to be *context-aware* to perceive and understand the situation, and *adaptive* to change its plan at run-time, allowing the user to answer the call. It may, for example, mute the TV and activate subtitles. In this case the smart home needed no user input, being *autonomous* to pursue goals to satisfy user needs.

In the beginning of the scenario, when the user asked to watch a movie, it is interesting that the smart home is capable to perform *goal reasoning* and infer secondary related goals, such as turning of lights and closing windows.

Another aspect we tackle is the fact that a smart home is constantly interacting with humans, therefore it must be *responsible*, i.e. in accordance with user values. In our

¹Such as <https://nodered.org>, and <https://ifttt.com>

²Process of finding alternative actions applicable in the current context

scenario, if the user privileges their leisure time more than their work, the smart home may decide to ignore the call if it was from the user's boss. But if family is of utmost importance, the system should never ignore a call from user's mom. This property goes beyond defining rules and constraints for system behavior [13], it dynamically reasons on the impact of user values on actions in current context.

The last property is *coordination*. The TV and the phone presented in the scenario must interact in order to perform any of the described adaptations.

It is important that all above properties are preserved during run-time, because smart home and ubiquitous systems in general must be constantly online and responsive.

3 Planning and Acting Agents

The previous section presented a global overview of the main motivation of this article, with special attention to smart homes. In these environments, we expect IoT objects to be intelligent agents, capable of acting upon and perceiving a dynamic world. This section will investigate the deliberation functions necessary to plan and act, and some state-of-the-art implementations. It is important to note that the word "deliberation" in this paper has the same meaning as in [14], which can sometimes differ from the common meaning we see in multi-agent systems literature.

The *adaptability*, *context-awareness* and *autonomy* of a smart home depends on the sophistication of acting and planning functions. The acting function receives the context from the external world (see Figure 1). The planning function above gives plans to be used by the acting one.

3.1 Acting

The "Acting" shown in Figure 1 is a deliberation function, the one that has direct contact to the external world, responsible for following plans (a set of ordered actions built to achieve goals), and react to context (with every input it checks if plans are still feasible and chooses appropriate available actions to be executed), all driven by previously chosen goals.

Among existing agent architectures, a popular one is the BDI (belief-desire-intention) agent model [15]. BDI agents have 3 components: *beliefs*, i.e. information about the current context; *desires*, i.e. goals the agent would like to achieve; *intentions*, i.e. agent's chosen goals to be achieved by executing a plan. BDI agents have a basic deliberation function that may be mapped to the "Acting" function described. The procedural plans building the plan library of BDI agent may be mapped to the "Know-How" appearing in Figure 1 in the sense that they define methods with their preconditions that can be used by the "Acting" function to execute and refine its execution based in context [14].

The BDI architecture inspired several agent-oriented programming languages, such as Jason [15], an implementation of AgentSpeak [16], technologies actively studied and expanded [17]. The BDI model has also been extended in many ways. For example, the suspension and resuming

of goals, not present in the original model, is presented in [18]. Also, a common goal notation called CAN (Conceptual Agent Notation) is proposed in [19].

Planning capabilities are not part of the BDI model. In Figure 1, the "Planning" step is executed before the agent becomes active – it means the agent will have access to a pre-compiled library of plans, but will not be able to create new plans during execution.

This behavior is similar to the Refinement Acting Engine (RAE) [14]. RAE-enabled agents present adaptive acting capabilities – they do not perform search algorithms to build new plans, but simply execute available plans applicable in the current context to achieve desired goals.

There are similar approaches in the AI field to create adaptive systems, such as [20], and others that use machine learning techniques, like neural networks, etc. We will focus on intelligent agents because they have an explainable process, whose reasoning is often easier to follow and comprehend than other AI techniques.

3.2 Planning

"Planning" is a deliberation function which complements "Acting" by providing it a set of plans customized to achieve previously defined user goals.

Several planning techniques exist, such as first-principles planning (FPP, informally, the creation of new plans based upon lower-level actions to achieve a goal) [21, 22], and markov decision process (MDP) [23]. Hierarchical Task Network (HTN) is a well-known planning technique in which abstract actions are refined into lower-level actions, until tangible actions that can be sent to execution are obtained [24]. In Figure 1, the HTN planner would be the "Planning" function. Its inputs are a set of user goals and an operational model (the "Know-What"), composed of more abstract actions than the commands in the "Know-How" set, which will be combined by the planner to create plans capable of achieving goals given a context. Its outputs are new plans. In this process, goals are the guiding elements of the planning process.

The difference between "Know-What" (*descriptive model* in [14]) and "Know-How" (*operational model* in [14]) in Figure 1 is: the former is a set of abstract actions used by the "Planning" function to create plans to achieve goals; the latter is a set of low-level commands used by the "Acting" to execute the abstract actions present in plans.

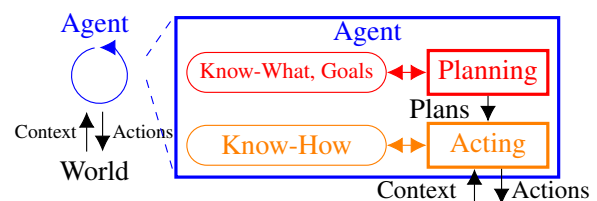


Figure 1: A planning and acting agent

4 Online Adaptation

Using planning and acting functions, described in the previous section, one can implement a basic agent, capable of planning by using known beliefs to create plans to react to predicted changes to achieve goals. But its "Planning" is not context-aware, therefore it provides plans compiled off-line, and does not adapt at execution since it has no feedback from "Acting" as shown in Figure 1.

To overcome the smart home challenge of being capable of adapting to context to satisfy user evolving needs, we thus seek to connect planning and acting capabilities. The expected synergistic result is an agent capable of performing online contextual and user-centric adaptation.

In our scenario from Section 2.3, if the smart home system never encountered the situation before (a call while playing a movie), it may need to perform planning in order to adapt to it. It may, for example, create a plan to pause the film and resume it when the call ends. By doing so, it performed online adaptation, a feature that highly improves system *adaptability*, *context-awareness* and *autonomy*.

4.1 Combining planning and acting

The original BDI model can be extended to perform lookahead planning. The integration of planning algorithms and agent reasoning was surveyed in [25]. An example is the embedding of FPP in BDI agents [21, 22], giving it some planning capabilities. Another is the use of HTN libraries as guides for BDI reasoning [26] aiming at combining HTN domain and BDI, allowing the latter to have access to the information contained in the former. A different approach fuses HTN planner with mental attitudes (beliefs, desires, and intentions) of the BDI architecture [27].

In a smart home, agents should do their best to autonomously adapt at run-time to user needs and environmental changes. Therefore we have interest in merging planning and acting capabilities, and even further, in a solution that perform both in a synergistic and online manner.

4.2 HTN planning and BDI acting

BDI agents and HTN planners are both well-known solutions. The former is capable of adaptive acting by interleaving refinement and execution of actions, and the latter performs planning in order to create new plans to achieve goals. The previous section showed that BDI agents can use the outputs (plans) from HTN planners, but Figure 1 clearly shows there is no feedback from the "Acting" function to the "Planning" one. It will prevent "Planning" from being context-aware, and the resulting agent will not be able to adapt to non-predicted situations.

If we establish that connection, as we can see in Figure 2, we create a feedback loop that allows the agent to perform online adaptation, i.e. online planning and acting. Now "Planning" has access to the current context, and can provide new plans to the "Acting" whenever it encounters a new situation, performing planning at run-time.

The Refinement Engine for Acting and Planning (REAP)

[14] and the BDI agent programming language named CANPlan [28, 29] are solutions that aim to incorporate HTN-style planning into BDI-like agents, ultimately allowing agents to perform online planning and acting.

Another solution is HTN Acting [30, 31], whose approach also combines HTN planning with BDI behavior, i.e. performing interleaved deliberation, acting and failure recovery. By adapting HTN planning semantics, HTN Acting does the opposite of REAP and CANPlan, that is adapting BDI agents.

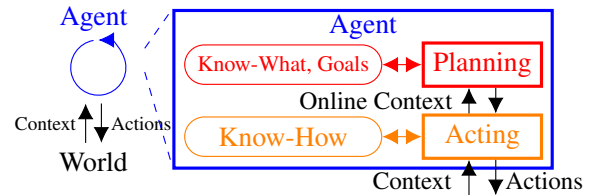


Figure 2: An online planning and acting agent

5 Responsible Goal-driven Adaptation

Agents in a smart home are subject to environmental changes, and the agent architecture presented in the previous section is capable of online adaptation to those changes. In this section, we identify that agent adaptation must also take into account user goals and values, two concepts that deserve specialized treatment during agent deliberation by the performing of goal- and value-reasoning.

These two reasoners are able to give a smart home the *goal-driven* and *responsible* properties we are looking for.

5.1 Beyond planning and acting

In the previously presented scenario in Section 2.3, we saw the system chose to achieve secondary related goals. The agent capabilities were beyond online planning and acting, its behavior were augmented to perform goal-reasoning. It gave the system a higher level of autonomy, improving user experience since it needed minimum to no user input.

The smart home from our scenario also presents value-reasoning capabilities, which increased system autonomy by helping the choice to answer or ignore the call from user's boss or mom. This function directly influences the goal-reasoning, the planning and the acting processes.

In the following sections we better elaborate the goal- and value-reasoning.

5.2 Goal-reasoning automation

In the literature we find two main types of goals [32]: *goals-to-be* (a.k.a. declarative goals) – i.e. states the agent wants to achieve or maintain [33]; and *goals-to-do* (a.k.a. procedural goals) – i.e. sets of actions the agent wants to successfully execute.

Goals are the guiding elements of the planning process. Most systems expect users to provide the goals to be pursued. In this case, they have automated planning and act-

ing, but they do not deliberate about which goals should be pursued, i.e. they do not reason upon goals, nor create new goals, nor choose which ones to pursue, nor drop existing ones (except in cases of failure or inability to achieve).

In our example scenario, the creation and activation of "turn off lights" and "close window" goals can be performed by a goal-reasoner as shown in Figure 3, which using its current beliefs (e.g. lights are on, windows are open, user started watching a movie) would reason and decide to pursue those goals to maximize user comfort, bettering user experience.

The architecture presented in [34, 35] is an example of a basic goal reasoning, which autonomously creates goals to redirect agent behavior if an user value is at stake. The work from [36] gives operational semantics in the CAN [19] language that enables goal reasoning features necessary to manage goals and plans, such as abortion, suspension and resuming. Another approach presents a goal deliberation strategy called Easy Deliberation [37], which allows activation and deactivation of goals in BDI agents, as well as defining relationships among goals to enable conflict free pursuing.

5.3 Value-driven responsibility

Smart home systems are designed for human users, and therefore they should be responsible systems by taking into account user values, which may include for example the user safety and privacy. The value-reasoning function shown in Figure 3 reasons based on current available beliefs, choosing to prioritize one value over another. In the scenario where a user receives a call while watching a movie, the system could prioritize work-related value over leisure-related value if, for example, the user calendar showed an important work meeting the next day, therefore if the boss was calling their call would not be ignored.

Figure 3 illustrates that the system reasoning process happens in the following order: first, it ranks the values based on the current context; second, it chooses goals that respect those values; third, it chooses or creates plans to achieve the chosen goals that respect the values; finally, it performs acting to realize those plans.

Value-based reasoning research have been highly influenced by The Theory of Basic Human Values [38], that identifies ten motivationally distinct values. For example, the article [39] performs value-reasoning by creating a constraint satisfaction problem in which all applicable plans are ranked using human values, and the plan that better suits user values is chosen. Note that the order of reasoning is the opposite of the one presented in Figure 3: here the agent first chose goals and create plans, then later ranks the available plans based in user values.

The architecture in Figure 3 has similarities to the emotional BDI agent architecture proposed in [34, 35], which includes not only values but also emotional appraisal into the agent reasoning, but whose architecture works differently: its goal reasoning is not affected by the values, in-

stead once the system interprets that the current context endangers user values, it will create a goal that tells the system to satisfy the endangered values.

Another approach use cultural values and rules to not only influence agent individual behavior, but to help coordinating it with other agents that, being in the same environment, have also similar cultural values and rules [40].

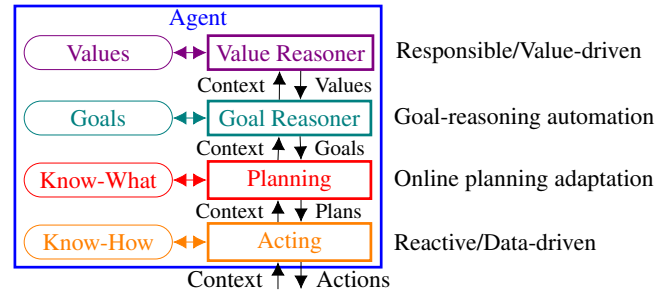


Figure 3: A responsible goal-driven online planning and acting agent

6 Multi-agent Adaptation

An agent with the architecture presented in the previous section is capable of adapting itself in an online, goal-driven and value-based fashion. But its behavior is completely ignorant of the presence of other intelligent agents, ultimately simply sensing other agents actions in the world and considering them simple changes to be adapted to. Therefore we are interested in allowing interaction and coordination among decentralised agents, as shown in Figure 4. We discuss in this existing solutions, their features and limitations for this coordination among agents.

6.1 Multi-agent system

Multi-agent systems (MAS) are compositions of multiple interacting intelligent agents. A smart home can be modeled as one. As proposed in [41], such systems are characterized by the presence of four dimensions: Agents, Environment, Interaction and Organisation.

The Agents dimension is composed of the intelligent agents themselves, i.e. entities capable of reasoning, perceiving and acting on their environment.

The Environment dimension is composed by artifacts the agents can interact with, i.e. real objects, digital services, etc. that can be used by agents in order to change the world or to have access to needed information.

The Interaction dimension constitutes the direct link among agents, allowing them to share messages.

The Organisation dimension establishes system norms and agents roles. The agents may or may not obey to system norms, but in a cooperative setting they would normally follow norms. Also, agents may have specific roles in the system, which gives them some responsibilities and consequently affects agents behaviors.

The exploration of these dimensions may help design a cooperative smart home comprising decentralised agents.

6.2 Coordination approaches

The architecture presented in [8] shows a possible approach using the organisation dimension as a mean of coordinating agents: the organisation has social schemes (sets of collective goals), which are used to allocate available agents that are capable of executing such goals. The allocation is done by giving roles to agents. In their system, the environment is a first-class abstraction [42], i.e. a component designed and programmed with clear responsibilities, such as providing agents with mechanisms for interaction and coordination and with access to available real objects.

Another approach, the Decentralised Online Multi-Agent Planning (DOMAP) presented in [43, 44], uses the environment dimension to coordinate agents. They have three special artifacts: "task board", "contract net board" and "social laws". The goals are declared in the "task board" artifact, and for each goal a "contract net board" is created, in which agents bid their "cost" for executing the goal. In the end, the agent with the best cost is chosen. The "social laws" are special artifacts that impose rules in order to resolve conflicts at run-time, and that all agents obey to.

Both presented approaches use a multi-agent programming framework called JaCaMo [45]. This framework allows the deployment of multi-agent systems by combining three platforms: Jason for the agents dimension, CArtaGo for the environment dimension, and Moise for the organisation dimension.

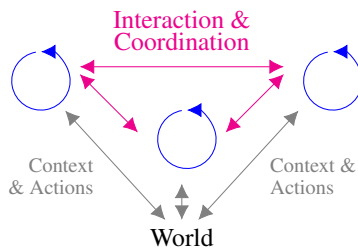


Figure 4: Decentralized agents coordinating, and perceiving and acting upon the world

7 Conclusions and Perspectives

The presented state-of-the-art insights help delimiting the current level of sophistication of each smart home identified requirements.

We saw that there are theoretical and concrete solutions that allow agents to perform online adaptation, i.e. online planning and acting, to a greater or lesser extent. For instance, the cited Refinement Engine for Acting and Planning (REAP), an augmented version of the Refinement Acting Engine (RAE), is a lead that is worth exploring to build a *context-aware*, *adaptive* and *autonomous* agent that plans and acts at run-time.

The BDI model has been shown to be highly expandable, e.g. incorporating planning capabilities. In future work we aim to develop the presented architecture in Figure 3, expanding even further the BDI model with *goal-driven* and

responsible behavior, allowing agents to interpret and take into account user goals and values.

However, as previously said, smart homes are composed of heterogeneous objects. Therefore, the next step following the design of an agent capable of responsible goal-driven online adaptation is to tackle the *coordination* challenge of decentralised agents, in a variety of IoT environments, including the smart home.

Such a multi-agent system may inspire future smart homes to be better designed to human users, by being not only intelligent, adaptive and coordinated systems, but also autonomous and trustworthy. The home of the future truly smart is hopefully just a few years away, and with help from the advancements in research, specially the emergence of AI and IoT solutions, the next human generation may start seeing their home as an integral part of their very own family.

References

- [1] M. Kovatsch, Y. N. Hassan, and S. Mayer, "Practical semantics for the internet of things: Physical states, device mashups, and open questions", in *5th International Conference on the Internet of Things, IOT 2015, Seoul, South Korea, 26-28 October, 2015*, IEEE, 2015, pp. 54–61.
- [2] F. Lécué, "Composition de services web: Une approche basée liens sémantiques. (web service composition : Semantic links based approach)", PhD thesis, École nationale supérieure des mines de Saint-Étienne, France, 2008.
- [3] D. Guinard, V. Trifa, T. Pham, and O. Liechti, "Towards physical mashups in the web of things", in *2009 Sixth International Conference on Networked Sensing Systems (INSS)*, Pittsburgh, PA, USA: IEEE, Jun. 2009, pp. 1–4. (visited on 01/14/2019).
- [4] M. Blackstock and R. Lea, "IoT mashups with the WoTKit", p. 8,
- [5] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*, Third edition, Global edition, in collab. with E. Davis and D. Edwards, ser. Prentice Hall series in artificial intelligence. Boston Columbus Indianapolis: Pearson, 2016, 1132 pp.
- [6] A. Ciorcea, O. Boissier, A. Zimmermann, and A. M. Florea, "Responsive decentralized composition of service mashups for the internet of things", in *Proceedings of the 6th International Conference on the Internet of Things, IOT 2016, Stuttgart, Germany, November 7-9, 2016*, S. Schneegass, A. Schmidt, F. Michahelles, M. Kritzler, A. Ilic, and K. Kunze, Eds., ACM, 2016, pp. 53–61.
- [7] M. Vallée, "Un intergiciel multi-agent pour la composition flexible d'applications en intelligence ambiante. (a multi-agent middleware for flexible application composition in ambient intelligence)", PhD

- thesis, École nationale supérieure des mines de Saint-Étienne, France, 2009.
- [8] A. Ciorcea, S. Mayer, and F. Michahelles, “Repurposing manufacturing lines on the fly with multi-agent systems for the web of things”, in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, E. André, S. Koenig, M. Dastani, and G. Sukthankar, Eds., International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 2018, pp. 813–822.
- [9] A. Sorici, G. Picard, O. Boissier, A. Zimmermann, and A. M. Florea, “CONSERT: Applying semantic web technologies to context modeling in ambient intelligence”, *Computers & Electrical Engineering*, vol. 44, pp. 280–306, 2015.
- [10] C. Perera, A. B. Zaslavsky, P. Christen, and D. Georgakopoulos, “Context aware computing for the internet of things: A survey”, *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
- [11] J. Euzenat, J. Pierson, and F. Ramparany, “Dynamic context management for pervasive applications”, *Knowledge Eng. Review*, vol. 23, no. 1, pp. 21–49, 2008.
- [12] A.-N. Ciorcea, “Weaving a social web of things : Enabling autonomous and flexible interaction in the internet of things. (tisser le web social des objets : Permettre une interaction autonome et flexible dans l’internet des objets)”, PhD thesis, École nationale supérieure des mines de Saint-Étienne, France, 2016.
- [13] P. Rust, G. Picard, and F. Ramparany, “Using message-passing DCOP algorithms to solve energy-efficient smart environment configuration problems”, in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, S. Kambhampati, Ed., IJCAI/AAAI Press, 2016, pp. 468–474.
- [14] M. Ghallab, D. S. Nau, and P. Traverso, *Automated Planning and Acting*. Cambridge University Press, 2016.
- [15] R. H. Bordini, J. F. Hübner, and M. Wooldridge, *Programming Multi-Agent Systems in AgentSpeak using Jason*. 2007.
- [16] A. S. Rao, “AgentSpeak(l): BDI agents speak out in a logical computable language”, in *Agents Breaking Away, 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Eindhoven, The Netherlands, January 22-25, 1996, Proceedings*, W. V. d. Velde and J. W. Perram, Eds., ser. Lecture Notes in Computer Science, vol. 1038, Springer, 1996, pp. 42–55.
- [17] A. Ricci, R. H. Bordini, J. F. Hübner, and R. Collier, “AgentSpeak(ER): Enhanced encapsulation in agent plans”, p. 16,
- [18] J. Thangarajah, J. Harland, D. N. Morley, and N. Yorke-Smith, “Suspending and resuming tasks in BDI agents”, in *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 1*, L. Padgham, D. C. Parkes, J. P. Müller, and S. Parsons, Eds., IFAAMAS, 2008, pp. 405–412.
- [19] M. Winikoff, L. Padgham, J. Harland, and J. Thangarajah, “Declarative & procedural goals in intelligent agent systems”, in *Proceedings of the Eight International Conference on Principles and Knowledge Representation and Reasoning (KR-02), Toulouse, France, April 22-25, 2002*, D. Fensel, F. Giunchiglia, D. L. McGuinness, and M.-A. Williams, Eds., Morgan Kaufmann, 2002, pp. 470–481.
- [20] A. Diaconescu, P. Mata, and K. Bellman, “Self-integrating organic control systems: From craysh to smart homes”, p. 8,
- [21] M. Xu, K. Bauters, K. McAreevey, and W. Liu, “A formal approach to embedding first-principles planning in BDI agent systems”, in *Scalable Uncertainty Management - 12th International Conference, SUM 2018, Milan, Italy, October 3-5, 2018, Proceedings*, D. Ciucci, G. Pasi, and B. Vantaggi, Eds., ser. Lecture Notes in Computer Science, vol. 11142, Springer, 2018, pp. 333–347.
- [22] L. d. Silva, S. Sardiña, and L. Padgham, “First principles planning in BDI systems”, in *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), Budapest, Hungary, May 10-15, 2009, Volume 2*, C. Sierra, C. Castelfranchi, K. S. Decker, and J. S. Sichman, Eds., IFAAMAS, 2009, pp. 1105–1112.
- [23] R. Bellman, “A markovian decision process”, *Journal of Mathematics and Mechanics*, vol. 6, no. 5, pp. 679–684, 1957.
- [24] A. Herzig, L. Perrussel, and Z. Xiao, “On hierarchical task networks”, in *Logics in Artificial Intelligence - 15th European Conference, JELIA 2016, Larnaca, Cyprus, November 9-11, 2016, Proceedings*, L. Michael and A. C. Kakas, Eds., ser. Lecture Notes in Computer Science, vol. 10021, 2016, pp. 551–557.
- [25] F. Meneguzzi and L. d. Silva, “Planning in BDI agents: A survey of the integration of planning algorithms and agent reasoning”, *Knowledge Eng. Review*, vol. 30, no. 1, pp. 1–44, 2015.

- [26] L. d. Silva, “BDI agent reasoning with guidance from HTN recipes”, in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2017, São Paulo, Brazil, May 8-12, 2017*, K. Larson, M. Winikoff, S. Das, and E. H. Durfee, Eds., ACM, 2017, pp. 759–767.
- [27] G. Wickler, S. Potter, A. Tate, M. Piechouicek, and E. Semsch, “Planning and choosing: Augmenting HTN-based agents with mental attitudes”, in *Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Silicon Valley, CA, USA, November 2-5, 2007*, IEEE Computer Society, 2007, pp. 222–228.
- [28] S. Sardiña, L. d. Silva, and L. Padgham, “Hierarchical planning in BDI agent programming languages: A formal approach”, in *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, H. Nakashima, M. P. Wellman, G. Weiss, and P. Stone, Eds., ACM, 2006, pp. 1001–1008.
- [29] S. Sardiña and L. Padgham, “A BDI agent programming language with failure handling, declarative goals, and planning”, *Autonomous Agents and Multi-Agent Systems*, vol. 23, no. 1, pp. 18–70, 2011.
- [30] L. d. Silva, “HTN acting: A formalism and an algorithm”, in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, E. André, S. Koenig, M. Dastani, and G. Sukthankar, Eds., International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 2018, pp. 363–371.
- [31] —, “Addendum to “HTN acting: A formalism and an algorithm””, *CoRR*, vol. abs/1806.02127, 2018.
- [32] M. B. v. Riemsdijk, M. Dastani, and M. Winikoff, “Goals in agent systems: A unifying framework”, in *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 2*, L. Padgham, D. C. Parkes, J. P. Müller, and S. Parsons, Eds., IFAAMAS, 2008, pp. 713–720.
- [33] S. Duff, J. Thangarajah, and J. Harland, “Maintenance goals in intelligent agents”, *Computational Intelligence*, vol. 30, no. 1, pp. 71–114, 2014.
- [34] N. Cointe, “Jugement éthique pour la décision et la coopération dans les systèmes multi-agents”, 2017.
- [35] C. Battaglino, R. Damiano, and L. Lesmo, “Emotional range in value-sensitive deliberation”, in *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013*, M. L. Gini, O. Shehory, T. Ito, and C. M. Jonker, Eds., IFAAMAS, 2013, pp. 769–776.
- [36] J. Harland, D. N. Morley, J. Thangarajah, and N. Yorke-Smith, “Aborting, suspending, and resuming goals and plans in BDI agents”, *Autonomous Agents and Multi-Agent Systems*, vol. 31, no. 2, pp. 288–331, 2017.
- [37] A. Pokahr, L. Braubach, and W. Lamersdorf, “A goal deliberation strategy for BDI agent systems”, in *Multiagent System Technologies*, T. Eymann, F. Klügl, W. Lamersdorf, M. Klusch, and M. N. Huhns, Eds., red. by D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, and G. Weikum, vol. 3550, Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 82–93. (visited on 04/01/2019).
- [38] S. H. Schwartz, “Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries”, in *Advances in Experimental Social Psychology*, vol. 25, Elsevier, 1992, pp. 1–65. (visited on 03/18/2019).
- [39] S. Cranefield, M. Winikoff, V. Dignum, and F. Dignum, “No pizza for you: Value-based plan selection in BDI agents”, in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, C. Sierra, Ed., ijcai.org, 2017, pp. 178–184.
- [40] L. Vanhée, *Using Cultures and Values to Support Flexible Coordination*. Utrecht University, 2015.
- [41] O. Boissier, R. H. Bordini, J. F. Hübner, and A. Ricci, “Dimensions in programming multi-agent systems”, *Knowledge Eng. Review*, vol. 34, e2, 2019.
- [42] D. Weyns, A. Omicini, and J. Odell, “Environment as a first class abstraction in multiagent systems”, *Autonomous Agents and Multi-Agent Systems*, vol. 14, no. 1, pp. 5–30, 2007.
- [43] R. C. Cardoso and R. H. Bordini, “A modular framework for decentralised multi-agent planning (extended abstract)”, p. 3, 2017.
- [44] R. C. Cardoso, “A decentralised online multi-agent planning framework for multi-agent systems”, 2018.
- [45] O. Boissier, R. H. Bordini, J. F. Hübner, A. Ricci, and A. Santi, “Multi-agent oriented programming with JaCaMo”, *Sci. Comput. Program.*, vol. 78, no. 6, pp. 747–761, 2013.

Viaduc : Un assistant de viabilité pour la négociation dans un éco-socio-système

Laetitia Zaleski¹ Isabelle Alvarez², Jean-Pierre Briot^{1&4} et Marta de Azevedo Irving³

¹ Sorbonne Université, CNRS, Laboratoire d'Informatique de Paris 6, LIP6, Paris

² Irstea, Paris

³ Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro

⁴ Universidade Federal do Estado do Rio de Janeiro (UNIRIO), Rio de Janeiro

zaleski@poleia.lip6.fr, isabelle.alvarez@irstea.fr, jean-pierre.Briot@lip6.fr, mirving@mandic.com.br

Résumé

Dans cet article, nous présentons notre projet de recherche qui vise à un soutien informatisé à la prise de décisions en matière de gestion collective participative des aires protégées. Ce projet consiste en un assistant pour la négociation permettant aux utilisateurs la formulation d'un projet cohérent, compatible avec ceux des autres utilisateurs, et facilitant la recherche de compromis. Nous proposons d'utiliser les concepts et outils de la théorie mathématique de la viabilité et nous illustrons notre approche avec un modèle de parc marin.

Mots-clefs

Viabilité, Aide à la décision, Gestion participative.

Abstract

In this paper, we introduce our current research project aimed at computer-based support for decision making in the participatory management of protected areas. This project consists of an assistant that allows users to formulate a self-consistent project, compatible with those of other stakeholders, and facilitates the search for compromises. We propose to use the concepts and tools of the mathematical viability theory and we illustrate our approach with a marine park model.

Keywords

Viability, decision-making, participatory management.

1 Introduction

Le tourisme consacré à la faune sauvage, c'est-à-dire la pratique de l'observation et de l'interaction avec la faune et la flore locales dans leurs habitats naturels, a augmenté au cours des dernières années, passant de 79 à 440 millions de participants (International Ecotourism Society, [1], Moorhouse et al. [2]) et devrait doubler au cours des 50 prochaines années (French et al.) [3]. Le tourisme de faune peut apporter des avantages financiers aux communautés locales tout en contribuant à la préservation de la biodiversité, en augmentant significativement le commerce régional (Ballantyne et Packer) [4]. Cependant, il est également impératif que ce tourisme lui-même soit géré efficacement pour s'assurer que les impacts négatifs ne l'emportent pas sur les effets positifs obtenus [5]. C'est pourquoi la gestion des aires protégées est un problème récurrent dans le domaine de la préservation de la biodiversité. Ces zones comptent de nombreuses ressources, ce qui peut entraîner des conflits entre les différents acteurs sociaux (ONG de protection des tortues, association de pêcheurs, opérateur de tourisme...). Nous reprenons l'analyse de Briot et al. [6] sur les différents acteurs et types de gestion, à laquelle nous ajoutons un assistant pour la négociation grâce à l'utilisation de la théorie de la viabilité, de manière individuelle et collective. Quelques exemples de conflits inhérents à la protection de la biodiversité sont : l'exploitation touristique inadéquate, la dégradation de l'environnement et l'utilisation illégale des ressources naturelles. Dans cet article, nous avons choisi de nous concentrer sur le tourisme de la faune marine et son impact sur les tortues marines au Brésil, car c'est l'une des formes les plus populaires et les plus répandues de tourisme marin [5]. Nous nous inspirons d'un parc marin situé à Arraial do Cabo, dans l'État de Rio de Janeiro au Brésil, où touristes,

tortues et pêcheurs interagissent. Nous décrivons d'abord ces interactions dans un modèle dynamique. De tels modèles ont été proposés pour la gestion des pêcheries par Otto et Struben [7] et par Videira [8] pour l'écotourisme, chacun se concentrant sur un aspect particulier (la pêche pour le premier et le tourisme écologique pour le second). Dans notre modèle nous souhaitons tenir compte de plusieurs aspects, afin de rendre compte de la potentielle présence de conflits. En s'inspirant des travaux de ComMod (Companion Modelling) nous ajoutons une dimension dynamique et un assistant pour la prise de décision [9]. Cependant, la dynamique du parc n'est pas simulée à l'aide de simulations multi-agent mais par un système d'équations, ce qui nous permet le calcul de noyaux de viabilité. Notre modèle s'inspire de ce parc marin mais n'est pas calibré sur ce parc. Il nous permet la construction d'un assistant à la négociation (voir Guyot et al.[10]), permettant des analyses comparatives de différents points de vue (en fonction des objectifs et des contraintes fixés par les différents utilisateurs), afin de faciliter une base de négociation non ambiguë. Dans cet article, nous décrivons le modèle et les outils mathématiques que nous avons utilisés, puis nous abordons en détail le fonctionnement de l'assistant et notre travail futur.

2 Modélisation du problème

Le modèle prend en compte l'évolution au cours du temps de la population animale (de tortues, notée $A(t)$), de l'activité touristique (le nombre de touristes noté $T(t)$), et du capital des infrastructures de pêches (noté $C(t)$). $(A(t), T(t), C(t))$ représente l'état du parc au temps t . L'évolution est décrite par un système d'équations différentielles. De plus nous considérons que les parties prenantes dans la gestion du parc souhaitent conserver le parc dans un ensemble d'états souhaitables $K \in \mathbb{R}^3_+$. Pour cela elles considèrent des moyens d'actions qui permettent de faire varier des paramètres de contrôles entre certaines bornes. Les touristes se rendent au parc marin pour (entre autres) observer ces tortues et la zone protégée est ouverte à la pêche. Les pêcheurs offrent des services aux touristes (excursions en bateau par exemple). Leur pêche conduit à la capture accidentelle de tortues marines. Pour définir le modèle, nous nous sommes inspirés du travail sur le tourisme durable de Casagrandi et Rinaldi [11]. Comme décrit dans la figure 1, les touristes (T) et le capital des pêcheurs (C) ont un impact négatif sur le nombre de tortues (A) (flèche annotée "-" sur la figure 1), mais le nombre de tortues et le capital des pêcheurs sont attractifs pour les touristes (flèche annotée "+" sur la figure 1). Le capital des infrastructures est indirectement influencé par le nombre de tortues car nous considérons que ce nombre est proportionnel au nombre de poissons, il s'agit de la flèche pointillée de la figure 1. Nous choisissons ici un modèle

simple, avec un nombre de variables réduit, afin que les calculs de viabilité décrit par la suite se fassent dans un temps raisonnable.

2.1 Les animaux

Le nombre d'animaux est calculé par l'équation :

$$\frac{dA}{dt} = g \left(A - \frac{A^2}{K_A} \right) - \zeta l T A - p A C i_p \quad (1)$$

avec $K_A = 1 + M [1 + \eta T / (1 + \epsilon)]^{-1}$.

La dynamique des animaux correspond à l'équation (1) qui est une loi de Verhulst [12] pour laquelle $g(1 - \frac{A(t)}{K_A})$ est le nombre de tortues qui seraient présentes sans interférence touristique directe. La capacité de charge K_A est la taille maximale de la population de tortues que le parc peut supporter, elle dépend de la capacité de charge absolue M (c'est à dire le nombre de tortues qu'il pourrait y avoir sans interférence humaine), ainsi que des dommages liés à la pollution touristique ηT et de l'effort pour restaurer l'environnement ϵ . La variable $A(t)$ inclut le nombre de tortues tuées par interaction directe avec les touristes : $\zeta l T(t) A(t)$ et celles prises accidentellement par les bateaux de pêche. Nous supposons que la pêche entraîne une mortalité accidentelle proportionnelle à l'activité de pêche, comme le font Peckham et al. [13]. Ce qui correspond dans l'équation 1 à $-p A C i_p$, où $C i_p$ est la valeur du capital investi dans les infrastructures de pêche. Les valeurs des paramètres correspondent aux valeurs trouvées dans la littérature. En raison du manque de données disponibles, certaines valeurs sont parfois extrapolées à partir des données sur d'autres espèces marines, elles aussi exposées au tourisme faunique. Les valeurs sans référence sont le fruit d'une analyse du modèle. Détaillons les valeurs de paramètres :

- Le taux de naissance n est estimé à 4.5% par an [14] et le taux de mortalité m est estimé à $m = 1\%$ par an [15]. Il en résulte $g = n - m = 3.5$ (Table 1).
- Le taux de mortalité des tortues (noté l) est lié à l'interaction directe avec les touristes [16].
- Le paramètre p correspond à la proportion de tortues tuées par la pêche.
- le paramètre M correspond à la capacité maximale des animaux [17].
- η correspond au nombre par lequel un touriste divise la capacité de l'environnement [18].
- ϵ correspond à l'effort pour restaurer l'environnement et ζ correspond à la proportion des espaces ouverts dans le parc. Ce sont les deux paramètres de contrôle et leur valeur est modifiable.
- i_p est la proportion du revenu de pêche qui est investie dans l'infrastructure de pêche.

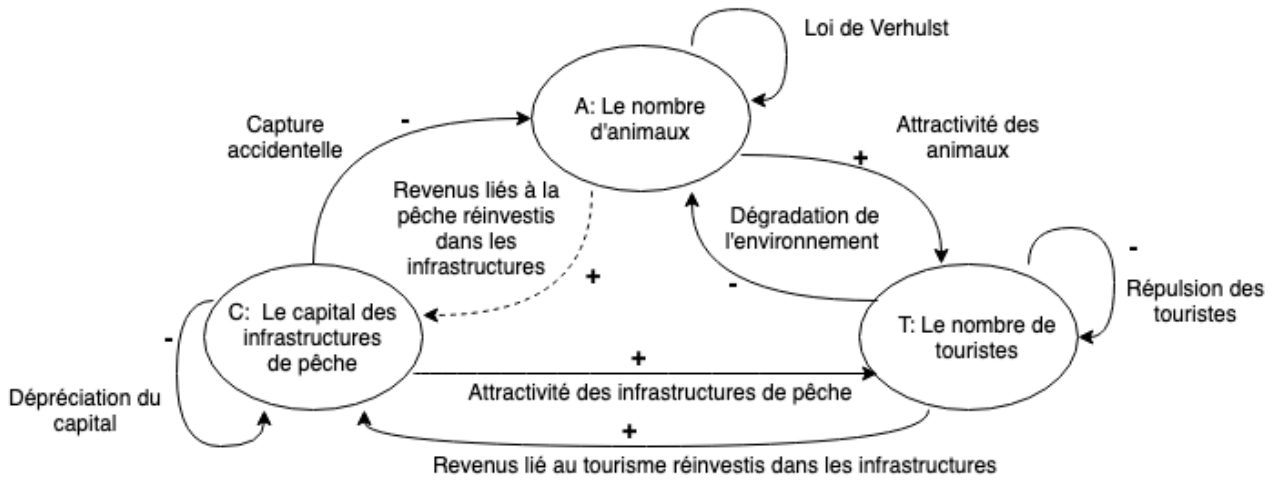


FIGURE 1 – Influence des variables les unes sur les autres

	g	l	p	M	η		δ	$pm_p i_p$	$m_t i_t$
Référence	[15, 16]	[17]	[17]	[18]		Référence	[19]	[21]	[22]
Valeur	0.035	0.0052	0.136	$3.6 \cdot 10^4$	$8 \cdot 10^{-4}$	Valeur	0.01	$3.3 \cdot 10^6$	$5 \cdot 10^6$
Unité	an^{-1}	an^{-1}	an^{-1}	Tortues	Touristes $^{-1}$	Unité	an^{-1}	Réal $\times \text{an}^{-1}$	Réal $\times \text{an}^{-1}$

TABLE 1 – Valeur des paramètres pour l'équation des animaux (1).

Les valeurs de ces paramètres sont donnés dans la table 1. Les paramètres ϵ et ζ sont des paramètres de contrôle ajustables et ne sont donc pas indiqués dans cette table.

2.2 Le Capital

Le capital des infrastructures de pêche est calculé par l'équation :

$$\frac{dC}{dt} = -\delta C + pm_p i_p A + m_t i_t T \quad (2)$$

Le capital suit l'équation (2), où $C(t)$ représente l'argent investit dans les infrastructures de pêche. Il diminue avec le temps d'un facteur de δ . Deux sources contribuent au capital : d'une part, les revenus de la pêche $pA(t)m_p i_p$ et, d'autre part, les revenus du tourisme : $T(t)m_t i_t$. Détaillons ces paramètres :

- $\delta = 0.01$ correspond à la dépréciation des infrastructures de pêche [19]
- le produit $pm_p i_p A$ correspond à l'argent provenant de la pêche [20], avec :
 - m_p est le prix du nombre de poissons capturés pour une tortue capturée
 - Comme dans l'équation (1) i_p est la proportion du revenu de pêche qui est investie dans l'infrastructure de pêche
- mt correspond à l'argent payé par les touristes aux pêcheurs [21] et [22]

TABLE 2 – Valeur des paramètres pour l'équation du capital (2)

— i_t est la proportion du revenu touristique qui est réinvestie dans les infrastructures

Les valeurs de ces paramètres sont données dans la table 2.

2.3 Les Touristes

Le nombre de touristes est calculé par l'équation :

$$\frac{dT}{dt} = -\alpha T^2 + \frac{a\zeta AT}{A + \phi} + \frac{\epsilon CT}{C + \phi_2 T + \phi_2} \quad (3)$$

Pour l'évolution du nombre de touristes, nous appliquons l'équation (3), inspirée de Casagrandi et Rinaldi [11]. Les touristes sont attirés par les animaux $\frac{a\zeta A(t)}{A(t) + \phi}$. Ils sont d'autant plus attirés s'ils sont plus susceptibles de voir un animal, mais jusqu'à un certain degré (ils ne seront pas deux fois plus attirés s'il peuvent voir deux tortues plutôt qu'une). Ceci est indiqué par le paramètre de saturation ϕ . Les touristes sont également attirés par les infrastructures de la même manière. De plus, les touristes se repoussent entre eux, ce qui est exprimé par le paramètre α . Le paramètre de congestion est $\alpha = 0.9$, il exprime le fait que les touristes préfèrent un faible nombre de touristes [11] et [23]. Détaillons ces paramètres :

- Le paramètre ϕ (valeur : 3666) est la contrainte de saturation demi-maximum, ce paramètre correspond au nombre de tortues pour lesquelles

la satisfaction des touristes est à la moitié du maximum.

- Le paramètre e est l'attractivité du lieu liée à la qualité des infrastructures des pêcheurs.
- Le paramètre ϕ_2 est la contrainte de saturation demi-maximum liée aux infrastructures de pêche.

Les valeurs de ces paramètres sont détaillées dans la table 3

Il est possible que les utilisateurs ne soient pas d'accord sur le modèle, mais on leur donne la possibilité de changer les paramètres s'ils le désirent. Ce modèle est déterministe. Les incertitudes sont prises en compte dans le cadre de la théorie mathématique de la viabilité qui s'intéresse au pire des cas.

2.4 Les paramètres de contrôle

Des actions de contrôle sont possibles et font partie intégrante de la modélisation. Le paramètre ζ représente la proportion du parc qui est ouverte au tourisme et le paramètre ϵ représente l'effort de restauration environnementale. Dans le modèle présenté, ζ et ϵ sont des paramètres de contrôle, ils ne sont pas fixes et peuvent varier dans une certaine plage. Pour $\zeta = 0$, ϵ ne change pas la valeur d'équilibre de A , C ou T . Pour des valeurs basses de ζ , ϵ ne change pas la valeur d'équilibre de C ou T , mais pour l'équilibre en A , des valeurs basses de ζ impliquent un effet plus fort du paramètre ϵ . Quand ζ est trop élevé ou trop bas, le nombre de touristes et le capital des infrastructures de pêche sont au plus bas.

3 La viabilité

3.1 Motivation

La théorie mathématique de la viabilité étudie la compatibilité entre un système dynamique et un ensemble de contraintes. Dans le problème que nous considérons, les acteurs sociaux définissent les contraintes acceptables et les moyens d'action. Nous utilisons dans la suite la théorie de la viabilité, introduite par Aubin en 1991 [24].

La théorie de la viabilité nous permet de considérer un système dynamique influencé par des paramètres de contrôle. Ces paramètres peuvent varier dans le temps afin de maintenir le système dans un ensemble de contraintes. Chaque participant peut décider quel sont les états souhaitables, en fonction de ses préférences. Par exemple, un pêcheur peut décider qu'il souhaite un capital des infrastructures de pêche supérieur à 50 000 Reals (monnaie brésilienne), un nombre de tortues entre 200 et 1000 individus et un nombre de touristes compris entre 5 000 et 20 000. L'ensemble souhaitable sera : $\{C \geq 50000, A \in [200, 1000], T \in [5000, 20000]\}$. Il choisira alors un ensemble de contrôles, par exemple : restauration

de l'environnement, de : "effort faible" à "effort élevé". Son contrôle suggéré sera alors : $\epsilon \in \{25, 50, 75\}$. En utilisant la théorie de la viabilité, l'assistant est capable de montrer au pêcheur s'il est possible de maintenir le système dans en utilisant les contrôles qu'il a suggéré. Plutôt que de définir des critères pour l'optimalité du système (par exemple, maximiser le nombre de tortues dans le parc), nous voulons ici considérer les contraintes sur les valeurs possibles de C , A et T , qui devraient être respectées pour que le système soit et demeure viable. La viabilité a déjà fait ses preuves en matière de développement durable [25]. Dans cet article, la viabilité nous sert également d'outil à la gestion participative et à la négociation. Un problème de viabilité peut être posé en terme de contrôle optimal mais pour des problèmes non linéaire et non quadratique la détermination de solution est difficile. Dans notre cas les participants n'utilisent pas nécessairement le même modèle, ici nous pouvons utiliser l'algorithme de Patrick Saint-Pierre [26] et les librairies existantes, comme Viabilitytree [27].

3.2 Le noyau de viabilité

La théorie de la viabilité considère un système dynamique défini par son état $\vec{x} \in X \subset \mathbb{R}^n$, dont l'évolution peut être influencée par un contrôle \vec{u} tel que :

$$\begin{cases} \vec{x}'(t) = f(\vec{x}(t), \vec{u}(t)) \\ \vec{u}(t) \in U(\vec{x}(t)) \end{cases}$$

où $U(\vec{x}(t)) \subset \mathbb{R}^q$ ($q \in \mathbb{N}$) est l'ensemble des contrôles admissibles et $\vec{u}(t)$ est une fonction de contrôle pour f .

Le noyau de viabilité est défini comme l'ensemble des états du système pour lesquels il existe toujours (au moins) une fonction de contrôle qui maintient le système dans les limites des contraintes, et est défini comme suit :

$$\text{Viab}(K) = \{\vec{x} \in K | \exists \vec{u}(\cdot) | \forall t \in [0, +\infty], \vec{x}'(t) \in K\}$$

L'ensemble K peut être défini par un système classique de contraintes statiques (par exemple $K = (x, y) | a \leq x \leq b, c \leq y \leq d$) La figure 2 montre un exemple de noyau de viabilité : pour tous les points de la zone en violet, il existe au moins une séquence de contrôle qui maintient la dynamique dans l'état souhaité.

4 L'assistant

4.1 Mode mono-utilisateur

L'assistant mono-utilisateur propose deux types d'aide. En premier lieu, l'aide à la cohérence du projet personnel. L'assistant permet à une personne de se rendre compte si ses moyens d'actions (représentés ici par des paramètres

	α	a	ϕ	ϕ_2	e
Référence	[11, 23]	\emptyset	\emptyset	\emptyset	\emptyset
Valeur	0.9	10000.0	3666	10000	100
Unité	Touristes $\times an^{-1}$	Animaux $\times an^{-1}$	Animaux $\times an^{-1}$	Réals $\times Touristes^{-1}$	Sans Unité

TABLE 3 – Valeur des paramètres pour l'équation des touristes (3)

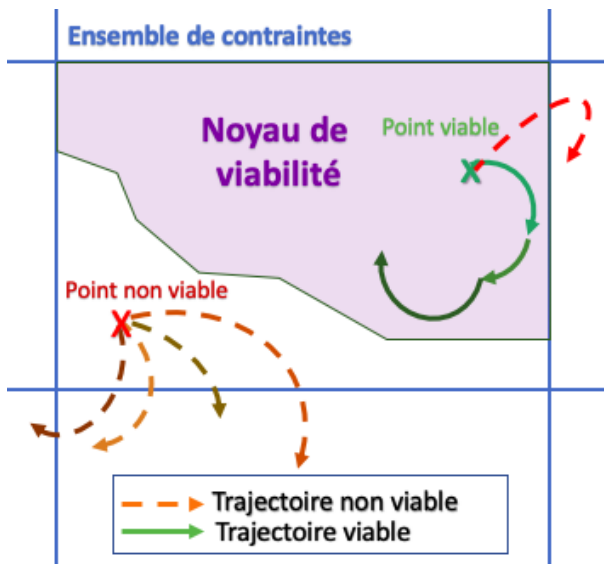


FIGURE 2 – Exemple d'un noyau de viabilité. La zone en violet correspond au noyau de viabilité. Les lignes pointillées sont les trajectoires non viables et les lignes pleines sont les trajectoires viables. Les lignes de différentes couleurs représentent des politiques de contrôles différentes. Plusieurs algorithmes permettent de calculer un noyau de viabilité [28] [29].

de contrôles) et ses contraintes sont compatibles avec le modèle en calculant un noyau de viabilité. Ce mode mono-utilisateur permet aussi dans un second temps de préparer l'utilisateur à la négociation qui sera faite dans le mode multi-utilisateur.

Le cas du noyau vide L'assistant peut être utilisé en mode mono-utilisateur afin de déterminer si le projet personnel de l'utilisateur est cohérent. Le projet est cohérent s'il est possible d'atteindre l'objectif de l'utilisateur étant données les hypothèses du modèle en utilisant les paramètres de contrôle choisis par ce dernier. Si les contraintes souhaitées ne sont pas compatibles avec la dynamique, ou si les intervalles de contrôle sont trop faibles, l'assistant calculera un noyau de viabilité vide. Cela signifie que l'éco-socio-système n'est pas viable sous les contraintes choisies.

Dans ce cas, l'utilisateur peut assouplir ses contraintes en en proposant un ensemble plus large ou rendre son action

plus efficace en ajoutant plus de possibilités de contrôle. L'assistant de viabilité peut aider les utilisateurs dans cette tâche en proposant de nouveaux ensembles de contrôles et de contraintes. Actuellement l'assistant propose le noyau correspondant au jeu de paramètres le plus proche parmi ceux stockés dans la base de données. Notre base de donnée contient à ce jour des noyaux calculés lors des précédentes utilisations de l'assistant. Dans le futur nous prévoyons une exploration systématique de l'espace des paramètres (contraintes et contrôles) à partir de suites de type Sobol [30] qui s'ajoutera au stockage des noyaux déjà calculés à la demande des utilisateurs. Les noyaux les plus proches sont déterminés à l'aide d'une distance euclidienne dans un espace à 10 dimensions (trois variables et deux paramètres de contrôles, possédant chacun une borne minimum et maximum). Il se peut qu'un joueur refuse d'utiliser un paramètre de contrôle, dans notre modèle, un utilisateur pourrait choisir d'utiliser seulement le paramètre de contrôle ϵ et de ne pas faire varier ζ par exemple. Dans ce cas on considère, pour le calcul de distance, que ses bornes minimum et maximum sont toutes deux égales à la valeur du paramètre.

Le cas du noyau non vide Si l'utilisateur a trouvé un noyau correspondant aux contraintes, l'assistant peut offrir de multiples informations sur ce noyau, et par extension, sur la difficulté à rester dans ce noyau.

L'une d'elles est l'analyse de sensibilité locale : en modifiant légèrement les différents paramètres du modèle, on peut dire si le noyau est sensible à la définition du problème de viabilité. Pour réaliser cette analyse de sensibilité, nous prévoyons d'utiliser une sous partie du plan factoriel déterminée par une matrice de Hadamard [30]. Si le volume ou la forme du noyau change beaucoup avec de petites modifications des paramètres, cela suggère un manque de robustesse du noyau par rapport à la définition du problème de viabilité, l'assistant pourra alors proposer un nouveau noyau, de manière similaire au cas du noyau vide.

Une autre fonctionnalité est de vérifier ce qui se passe si on ne veut pas s'approcher de la frontière du noyau. En effet, une fois que le système arrive près de la frontière du noyau, il devient plus difficile à contrôler (Alvarez et al. 2011 [31]). Il est possible de calculer un noyau érodé, c'est à dire un ensemble compris dans le noyau de viabilité, et pour lequel on interdit aux trajectoires de passer près

de la frontière du noyau. Si le volume du nouveau noyau érodé est trop faible par rapport au noyau précédent, cela suggère un manque de robustesse du noyau par rapport à la définition du problème de viabilité. Dans ce cas, l'assistant proposera une définition plus robuste. La figure 3 présente une coupe de noyau de viabilité et son érosion avec le logiciel Viabilitree [27].

Pour un état initial qui n'est pas dans le noyau de viabilité, on peut regarder s'il est possible de faire évoluer le système pour l'y ramener. Nous pouvons calculer le bassin de capture. Le bassin de capture du noyau de viabilité rassemble tous les états à partir desquels il existe au moins une suite d'actions (représentés ici par les paramètres de contrôles) qui permet d'atteindre le noyau. Le bassin de capture est représenté dans la figure 4 en orange. Le noyau de viabilité, en violet sur la figure 4, est compris dans le bassin de capture. Si un point est n'appartient pas à l'ensemble de contrainte (rectangle bleu sur la figure 4) mais qu'une suite d'action (ici un changement de paramètre de contrôle) permet d'atteindre le noyau alors ce point appartient au bassin de capture. C'est le cas du point en vert sur la droite de la figure 4, la trajectoire qui permet à partir de ce point d'atteindre le noyau est représenté en marron.

Pour illustrer les relations entre le noyau et la dynamique, l'assistant peut montrer à l'utilisateur quelques trajectoires viables. Lorsque le noyau est calculé, l'assistant stocke une stratégie de contrôle qui permet à la dynamique de rester dans les limites des contraintes choisies. En changeant les paramètres de contrôle en fonction de la politique calculée, il est possible de visualiser une trajectoire dynamique possible et sa politique de contrôle associée, en s'appuyant sur le système Vino [33].

Arguments à l'appui de la négociation dans un cadre multi-utilisateurs. En mode mono-utilisateur, l'utilisateur a la possibilité de calculer plusieurs noyaux pour se préparer à la négociation à venir. Après avoir calculé son noyau durant l'étape précédente, un utilisateur peut calculer un autre noyau en fonction de ce qu'il suppose être les contraintes et contrôles d'un autre participant. Ce nouveau noyau calculé permet à l'utilisateur d'obtenir de multiples informations telles que la présence ou l'absence d'intersection entre les noyaux. Ensuite, l'utilisateur peut vérifier si, selon ce qu'il pense être le contrôle d'un autre utilisateur, ils auront des paramètres de contrôles viables en commun.

Enfin, il peut vérifier s'ils partageront la même dynamique. Si deux personnes ne partagent pas la même dynamique, il leur est toujours possible de s'entendre sur une politique de contrôle. Pour ce faire, l'assistant calculera le *noyau garanti* [34], une fonctionnalité qui n'est pas encore implémentée. Le noyau garanti rassemble les états pour lesquelles il existe une politique de contrôle viable en

présence de perturbation appartenant à un ensemble. Les différentes valeurs de paramètres de la dynamique peuvent être vues comme des perturbations du modèle initial. La figure 5 représente un noyau garanti pour deux modèles différents, qui peuvent appartenir chacun à deux utilisateurs différents de l'assistant. Pour qu'un point de l'ensemble de contrainte appartienne au noyau garanti, il faut qu'il existe une suite d'actions (ici représentés par des paramètres de contrôles) tel que les deux modèles soient maintenus dans l'état souhaité. Si l'on commence nos calculs à partir du point vert en haut de la figure 5 et que l'on fixe nos paramètres de contrôle à la même valeur pour les deux modèles, nous obtenons deux résultats différents selon le modèle. Sur la figure 5, les trajectoires correspondant à des modèles différents sont représentées par des flèches de couleur différentes. Si les résultats appartiennent tous deux à l'ensemble de contraintes alors nous pouvons réitérer cette étape, c'est à dire, pour nos deux résultats obtenus, vérifier s'il existe un contrôle permettant de maintenir les deux modèles dans l'état souhaité.

Si deux personnes n'ont pas les mêmes variables de contrôle, l'assistant vérifiera d'abord s'il est possible de trouver un noyau en utilisant seulement l'intersection des contrôles. Dans le cas contraire, l'assistant utilisera l'union des contrôles, et les participants pourront discuter de cette politique de contrôle pendant la phase de négociation.

Bien que l'assistant soit pensé pour être utilisé en mode multi-utilisateur et pour la négociation entre plusieurs personnes, il est à ce jour uniquement implémenté en mode mono-utilisateur et ne comporte pas la possibilité de calcul de noyaux garantis ou de bassin de capture, qui seront ajoutés prochainement.

4.2 Mode multi-utilisateur

L'assistant est pensé pour pouvoir être utilisé en mode multi-utilisateur, permettant aux participants de partager des informations sur leurs contraintes et leurs noyaux. Dans ce mode, l'assistant pourra donner de multiples conseils pour la négociation future entre les participants. Afin de savoir s'il est possible de trouver une politique de contrôle sur laquelle tous les participants sont d'accord, nous procédons en plusieurs étapes. Nous devons d'abord vérifier si les contraintes des participants se recoupent. Si c'est le cas, nous pouvons vérifier si les noyaux des participants ont une intersection non vide. Dans le cas d'une intersection vide entre les différents noyaux des participants, au moins un d'entre eux devra changer son problème de viabilité.

En cas d'intersection non vide entre les noyaux de différents participants. Il est possible de calculer un nouveau noyau de viabilité dans cette intersection. Si les

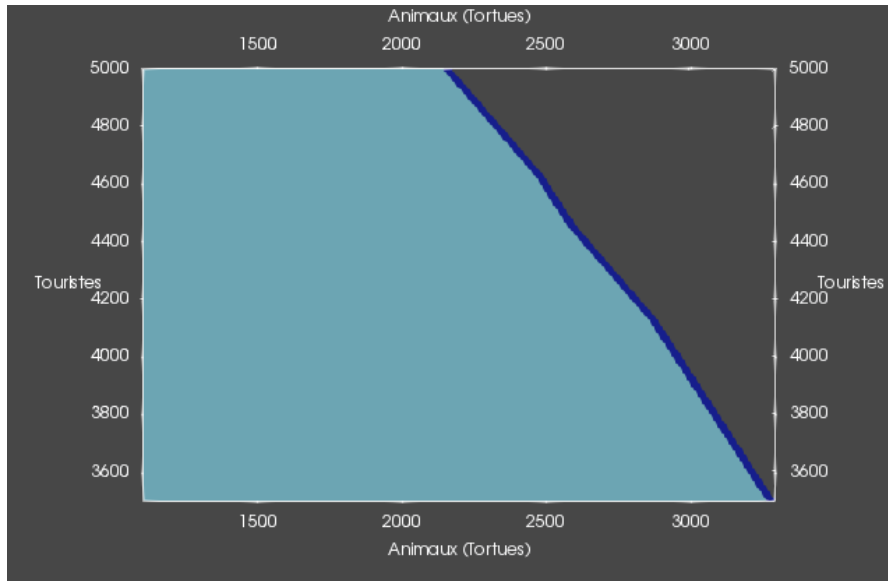


FIGURE 3 – Superposition de vues Animaux (abscisse) / Touristes (ordonnée) d’un noyau érodé (en bleu clair et au premier plan) et non érodé (en bleu foncé et en arrière plan). Figure réalisée avec Paraview [32]

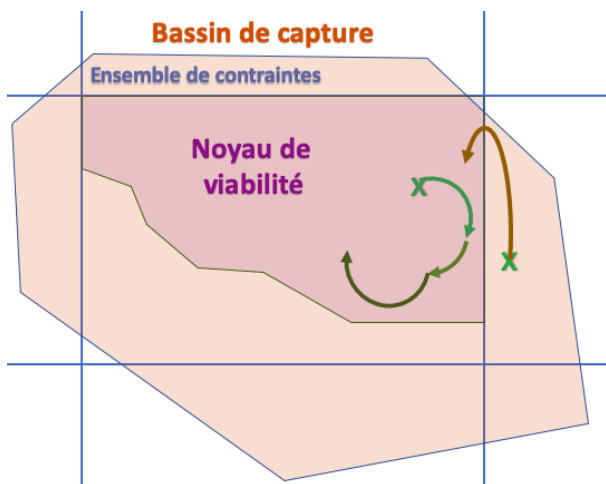


FIGURE 4 – Schéma du bassin de capture : les points verts appartiennent au bassin de capture et les flèches représentent les trajectoires associées à la dynamique.

utilisateurs ont le même modèle et les mêmes contrôles, tous les participants auront le même nouveau noyau, leur ensemble de contraintes étant désormais le même : l’intersection entre les noyaux de différents participants. Si le noyau est vide, certains participants devront reformuler leur problème de viabilité en modifiant les contraintes établies, les paramètres de contrôle ou les deux. Si le noyau n’est pas vide, cela signifie qu’un ensemble d’états communs a été trouvé. Sur la figure 6, nous observons sur la première ligne deux scénarios possibles : dans

la sous-figure de gauche les contraintes des différents participants (représentées par des rectangles jaunes, bleu et rouge) n’ont aucune intersection en commun. Dans ce cas, les participants sont invités à modifier leurs contraintes. Dans la figure de droite les noyaux des différents participants ont une intersection non-vide et il est donc possible de calculer un nouveau noyau à l’intérieur de cette intersection, comme montré dans la sous figure du bas. Dans cette dernière sous-figure, on observe que le noyau issu de l’intersection est différent de l’intersection des noyaux. Cela vient du fait que la viabilité n’est pas stable par rapport à l’intersection.

Si les utilisateurs appliquent une politique de contrôle différente, l’assistant peut calculer, pour l’union de tous les contrôles des utilisateurs, quelle politique de contrôle peut être utilisée afin que la trajectoire de la dynamique reste dans l’intersection calculée. Le nouveau noyau aura alors plus de paramètres de contrôle, ou bien une plage plus large de valeurs de paramètres de contrôles, par rapport aux problèmes de viabilité de certains des utilisateurs. Dans le cas où les participants considèrent une dynamique différente (si certains ont changé les paramètres du modèle par exemple), il est possible de ramener le problème de viabilité à un problème de noyaux garantis, comme dans le cas de la recherche d’argument de l’assistant individuel. Ces étapes sont illustrées sur la figure 7.

En cas d’échec aux étapes précédentes, l’assistant offre deux types d’aide : l’aide au choix de l’utilisateur qui devra assouplir ses contraintes et l’aide à l’assouplissement de contraintes

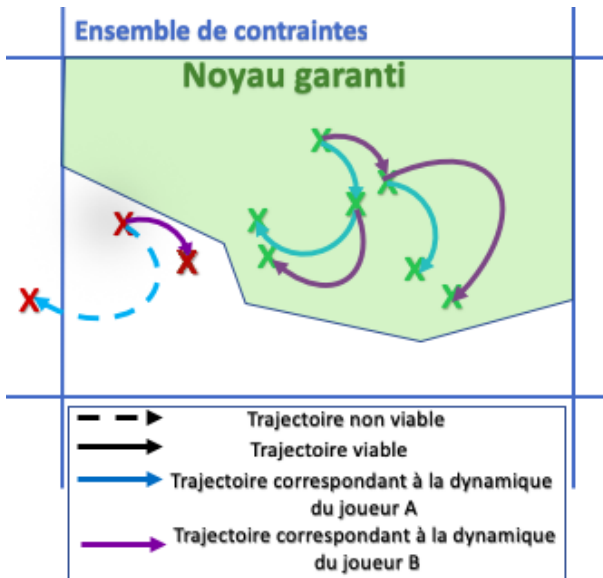


FIGURE 5 – Schéma de noyau garanti dans un scénario de jeu avec deux joueurs A et B. Les points en verts appartiennent au noyau garanti, les points en rouge n’y appartiennent pas.

Aider à l’assouplissement de contraintes En cas d’échec lors des étapes précédentes, de multiples indications peuvent être données aux participants afin de les aider à décider lequel d’entre eux devrait reformuler son problème de viabilité. L’assistant peut calculer le volume des noyaux pour déterminer lequel est le plus petit, et donner cette information aux utilisateurs. Cette personne peut alors choisir d’assouplir ses contraintes et contrôles afin que le noyau associé s’élargisse jusqu’à ce qu’une intersection non vide soit trouvée.

Une autre information que l’assistant peut donner aux participants est le noyau ayant le moins d’intersections non vides avec d’autres noyaux (dans le cas de trois participants ou plus). L’assistant compare chaque noyau deux à deux et calcule leur intersection, afin de déterminer quels noyaux ont conduit à l’intersection vide. A l’heure actuelle cette fonctionnalité est implémentée dans le mode mono-utilisateur.

Aider au choix de l’utilisateur qui devra assouplir ses contraintes Une fois que tous les participants se sont mis d’accord sur l’utilisateur qui devrait modifier son problème de viabilité, l’assistant peut l’aider à assouplir ses contrôles et ses contraintes. Si l’ensemble de contraintes est significativement plus grand que le noyau résultant, l’assistant conseillera d’assouplir les contrôles au lieu des contraintes. Par contre si la zone du noyau est proche de la zone de contrainte, l’assistant peut conseiller de relâcher les contraintes. Si l’utilisateur souhaite assouplir

les contraintes, l’assistant peut l’aider à décider de quelle manière les contraintes doivent être assouplies afin de créer une intersection non vide avec les noyaux des autres participants.

5 Conclusion

Dans cet article, nous avons proposé d’appliquer la théorie de la viabilité pour l’aide à la gestion. Bien que l’assistant s’appuie sur notre modèle de parc marin, inspiré d’une réserve naturelle brésilienne, il est pensé pour pouvoir être utilisé avec d’autres modèles sans que les étapes ne changent. La viabilité permet aux utilisateurs d’avoir une base commune et non ambiguë pour la négociation. A l’heure actuelle un prototype mono-utilisateur a été implémenté et testé sans joueurs. Les modules d’érosion et d’aide à la relaxation des contraintes sont en cours d’implémentation. Nous projetons d’utiliser la plate-forme OpenMole [35] et plus spécifiquement la méthode PSE (Pattern Space Exploration) pour l’exploration de noyaux afin de proposer des noyaux mieux adaptés. Cette méthode nous permettra d’explorer la diversité des résultats d’un modèle. Nous prévoyons également d’intégrer l’assistant dans un jeu sérieux de gestion de parc naturel [6].

Références

- [1] E. Society, “Ecotourism statistical fact sheet,” 1998.
- [2] T. P. Moorhouse, C. A. Dahlsjö, S. E. Baker, N. C. D’Cruze, and D. W. Macdonald, “The customer isn’t always right—conservation and animal welfare implications of the increasing demand for wildlife tourism,” *PloS one*, vol. 10, no. 10, p. e0138939, 2015.
- [3] S. S. French, M. González-Suárez, J. K. Young, S. Durham, and L. R. Gerber, “Human disturbance influences reproductive success and growth rate in California sea lions (*Zalophus californianus*),” *PloS one*, vol. 6, no. 3, p. e17686, 2011.
- [4] R. Ballantyne and J. Packer, *International handbook on ecotourism*. Edward Elgar Publishing, 2013.
- [5] C. Trave, J. Brunnschweiler, M. Sheaves, A. Diedrich, and A. Barnett, “Are we killing them with kindness? Evaluation of sustainable marine wildlife tourism,” *Biological conservation*, vol. 209, pp. 211–222, 2017.
- [6] J.-P. Briot, M. de Azevedo Irving, J. E. Vasconcelos Filho, G. M. de Melo, I. Alvarez, A. Sordoni, and C. J. P. de Lucena, “Participatory management of protected areas for biodiversity conservation and social inclusion : Experience of the SimParc multi-agent-based serious game,” in *Multi-Agent-Based Simulations Applied to Biological*

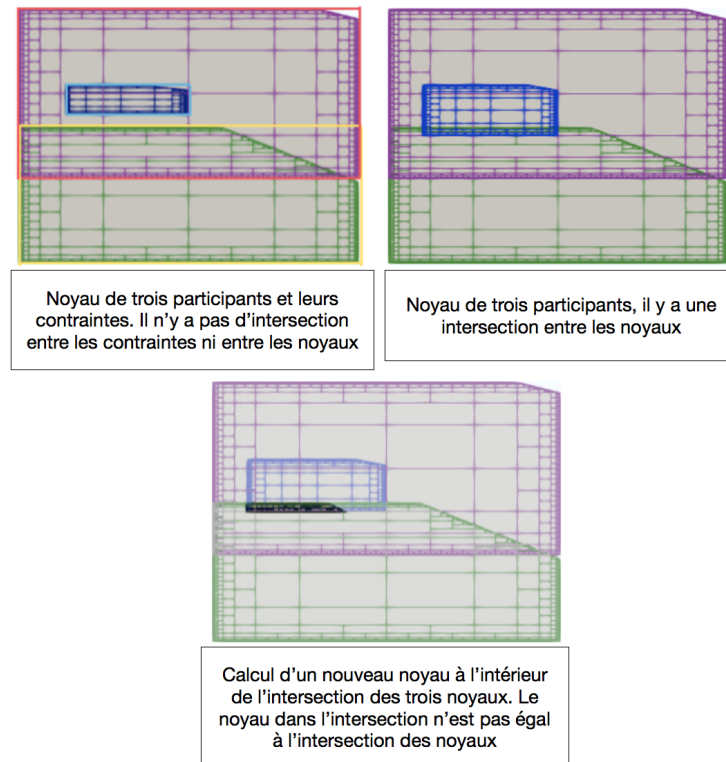


FIGURE 6 – Noyaux de trois participants dans des scénarios différents

- and *Environmental Systems*, pp. 295–332, IGI Global, 2017.
- [7] P. Otto and J. Struben, “Gloucester fishery : insights from a group modeling intervention,” *System Dynamics Review : The Journal of the System Dynamics Society*, vol. 20, no. 4, pp. 287–312, 2004.
- [8] A. Beall and L. Zeoli, “Participatory modeling of endangered wildlife systems : Simulating the sage-grouse and land use in central washington,” *Ecological Economics*, vol. 68, no. 1-2, pp. 24–33, 2008.
- [9] C. Le Page, *Simulation multi-agent interactive : engager des populations locales dans la modélisation des socio-écosystèmes pour stimuler l'apprentissage social*. PhD thesis, CIRAD-GREEN, 2017.
- [10] P. Guyot and S. Honiden, “Building coalitions involving agents and humans : reports from agent-based participatory simulations,” in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, p. 210, ACM, 2007.
- [11] R. Casagrandi and S. Rinaldi, “A theoretical approach to tourism sustainability,” *Conservation ecology*, vol. 6, no. 1, 2002.
- [12] M. Schtickzelle, “Pierre-François Verhulst (1804-1849). La première découverte de la fonction logistique,” *Population (French Edition)*, vol. 36, no. 3, pp. 541–556, 1981.
- [13] S. H. Peckham, D. M. Diaz, A. Walli, G. Ruiz, L. B. Crowder, and W. J. Nichols, “Small-scale fisheries bycatch jeopardizes endangered Pacific loggerhead turtles,” *PloS one*, vol. 2, no. 10, p. e1041, 2007.
- [14] J. R. Hendrickson, “The green sea turtle, *Chelonia mydas* (linn.) in Malaya and Sarawak,” in *Proceedings of the Zoological Society of London*, vol. 130, pp. 455–535, Wiley Online Library, 1958.
- [15] R. J. Brooks, G. P. Brown, and D. A. Galbraith, “Effects of a sudden increase in natural mortality of adults on a population of the common snapping turtle (*Chelydra serpentina*),” *Canadian Journal of zoology*, vol. 69, no. 5, pp. 1314–1320, 1991.
- [16] C. A. Semeniuk, W. Haider, A. Cooper, and K. D. Rothley, “A linked model of animal ecology and human behavior for the management of wildlife tourism,” *Ecological Modelling*, vol. 221, no. 22, pp. 2699–2713, 2010.
- [17] B. J. Godley, A. C. Broderick, and G. C. Hays, “Nesting of green turtles (*Chelonia mydas*) at ascension island, South Atlantic,” *Biological Conservation*, vol. 97, no. 2, pp. 151–158, 2001.
- [18] L. Bejder, A. Samuels, H. Whitehead, N. Gales, J. Mann, R. Connor, M. Heithaus, J. Watson-Capps,

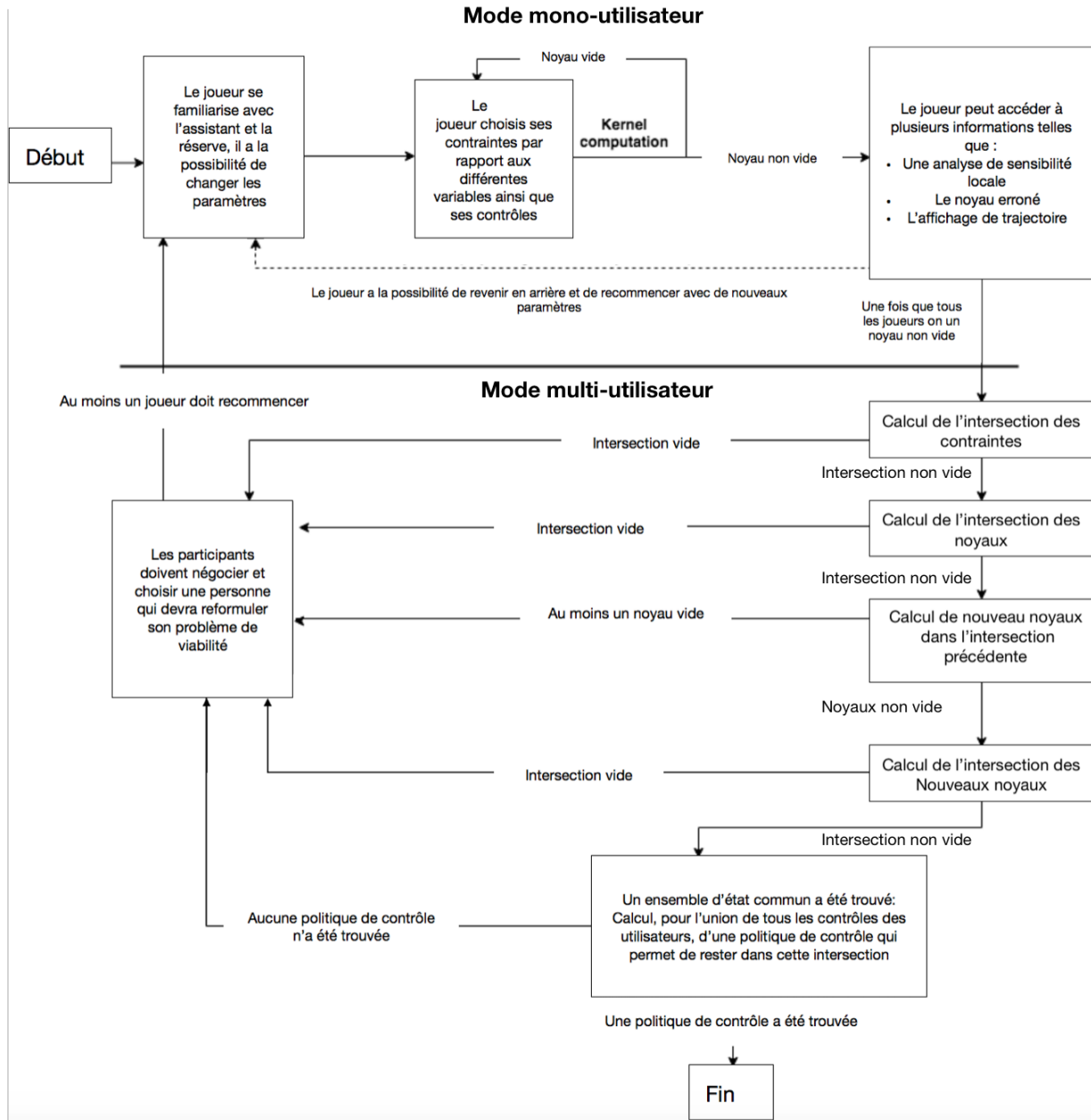


FIGURE 7 – Les différentes étapes de l'assistant

- C. Flaherty, and M. Krutzen, "Decline in relative abundance of bottlenose dolphins exposed to long-term disturbance," *Conservation Biology*, vol. 20, no. 6, pp. 1791–1798, 2006.
- [19] B. S. Lee, "Measurement of capital depreciation within the japanese fishing fleet," *the Review of Economics and Statistics*, pp. 225–237, 1978.
- [20] F. M. de Mendonça, R. Valle, and R. Coutinho, "A cadeia produtiva da pesca artesanal em Arraial do Cabo : análise e propostas de melhoria," *Liandra Peres Caldasso Rogério Valle*, p. 45.
- [21] Anonymous, "Fipac presta contas em audiência pública e ouve reivindicações de pescadores.," august 2018. <https://www.fiquebeminformado.com.br/2018/08/fipac-presta-contas-em-audiencia.html>.
- [22] V. G. da Vinha, P. May, and L. P. Caldasso, "Sustentabilidade da reserva extrativista marinha de Arraial do Cabo, rj : técnicas de pesquisa e resultados,"
- [23] J. Catlin and R. Jones, "Whale shark tourism at ningaloo marine park : a longitudinal study of wildlife tourism," *Tourism Management*, vol. 31, no. 3, pp. 386–394, 2010.
- [24] J.-P. Aubin and H. Frankowska, "Viability kernel of control systems," in *Nonlinear synthesis*, pp. 12–33, Springer, 1991.
- [25] C. Bernard, *La théorie de la viabilité au service de la modélisation mathématique du développement durable. Application au cas de la forêt humide de Madagascar*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II, 2011.
- [26] P. Saint-Pierre, "Approximation of the viability kernel," *Applied Mathematics and Optimization*, vol. 29, no. 2, pp. 187–209, 1994.
- [27] I. Alvarez, R. Reuillon, and R. De Aldama, "Viabilitree : A kd-tree framework for viability-based decision," 2016.
- [28] P. Saint-Pierre, "Approximation of the viability kernel," *Applied Mathematics and Optimization*, vol. 29, no. 2, pp. 187–209, 1994.
- [29] W. Wei, I. Alvarez, and S. Martin, "Algorithme d'approximation du noyau de viabilité avec procédure de classification," in *RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle)*, 2012.
- [30] A. Saltelli, *Global Sensitivity Analysis*. Wiley, 2008.
- [31] I. Alvarez and S. Martin, *Geometric Robustness of Viability Kernels and Resilience Basins*, pp. 193–218. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011.
- [32] J. Ahrens, B. Geveci, and C. Law, "Paraview : An end-user tool for large data visualization," *The visualization handbook*, vol. 717, 2005.
- [33] "Vino." <http://motive.cemagref.fr/vino/>.
- [34] J. P. Aubin, *Dynamic economic theory : a viability approach*, vol. 5. Springer Verlag, 1997.
- [35] R. Reuillon, M. Leclaire, and S. Rey-Coyrehourcq, "OpenMOLE, a workflow engine specifically tailored for the distributed exploration of simulation models," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 1981–1990, 2013.

III - Articles courts

Interface d'Aide à la Décision pour Prédire l'Hospitalisation de Patients et planifier les Traitements pour Prévenir cet Événement

Raphaël Gazzotti¹⁴

Elisa Noual⁴

Catherine Faron-Zucker¹

Fabien Gandon²

Alain Giboin²

Virginie Lacroix-Hugues³

David Darmon³

¹ Université Côte d'Azur, Inria, CNRS, I3S
nom@i3s.unice.fr

² Inria, Université Côte d'Azur, CNRS, I3S
prénom.nom@inria.fr

³ Université Côte d'Azur, Département de Médecine Générale
vhugues@outlook.fr, david.darmon@unice.fr

⁴ SynchroNext
prénom@synchronext.com

Résumé

Les médecins sont confrontés à une augmentation constante du nombre de leurs patients compte tenu du numerus clausus imposé aux études médicales en France. De plus, le vieillissement global de la population les oblige à traiter des patients atteints de nombreuses maladies chroniques (comorbidité), ce qui complique la prise en charge des patients puisque la polymédication implique l'apparition d'effets indésirables.

Dans cet article, nous présentons l'interface d'un algorithme développé pour aider à la prise de décision des médecins généralistes qui leur permet d'identifier les premiers signes qui mènent les patients à l'hospitalisation et les problèmes médicaux à traiter en priorité.

Mots Clef

Interface homme-machine, Dossier médical électronique, Modèle prédictif.

Abstract

Physicians are confronted with a constant increase in the number of their patients given the numerus clausus imposed on medical studies in France. In addition, the overall aging of the population requires them to treat patients with many diseases (comorbidity), which complicates the management of patients since polypharmacy implies the appearance of unexpected adverse drug's effects.

In this paper, we present the interface of an algorithm developed to assist in the decision-making process of general practitioners (GPs) that allows them to identify in patients

the first signs that lead to hospitalization and medical problems to be treated as a priority.

Keywords

Human-machine interface, Electronic medical record, Predictive model.

1 Introduction

Chaque année en France, le taux d'hospitalisation est 191 pour 1000 habitants (12,7 millions de patients pour plus de 3300 établissements de santé), aussi, une hospitalisation complète dans le secteur public dure en moyenne 6,4 jours en médecine, chirurgie, obstétrique, 36,4 jours en suivi et réadaptation, 44,5 jours en hospitalisation à domicile et 57,0 jours en psychiatrie.¹

Les médecins généralistes sont responsables des soins de 90% des problèmes de santé de la population et la multitude d'informations et la constante augmentation des patients rencontrés compliquent leurs tâches, c'est pourquoi il est nécessaire de leur offrir un résumé des dossiers patients incluant tous les points essentiels dans ce dernier et d'indiquer les actions thérapeutiques à choisir en priorité.

La base de données PRIMEGE PACA [1] qui contient plus de 440 000 consultations en français réalisées par 17 médecins généralistes nous sert de champ d'expérimentation pour construire une application qui aide un généraliste dans l'exercice de ses fonctions. Dans cette base de données, les descriptions textuelles rédigées par les médecins généralistes sont disponibles avec les codes de classification in-

1. <https://www.atih.sante.fr/analyse-de-1-activite-hospitaliere-2017>

ternationaux des médicaments prescrits, pathologies et raisons de consultation, ainsi que les valeurs numériques des différents résultats d'examen médicaux obtenus par un patient.

L'objectif d'une telle application est double puisqu'elle permet à la fois d'améliorer l'état de santé des patients en identifiant d'abord les patients à haut risque et potentiellement d'éviter leurs hospitalisations en affichant les facteurs de risque classés en fonction de leur participation à la décision de l'algorithme.

Par ailleurs, l'absence de recommandations spécifiques pour le traitement des patients polyopathologiques² complique la prise en charge des patients avec le risque d'appliquer les recommandations pour les maladies isolées [2, 3], ce qui confirme l'intérêt pour le médecin généraliste de pouvoir identifier les éléments à prendre en charge en priorité.

2 Contexte et travaux connexes

Les auteurs de [4] expriment que l'adoption des systèmes exploitant les dossiers médicaux électroniques (DME) peut réduire considérablement les coûts dans le domaine des soins de santé, de plus, ces systèmes peuvent être utilisés afin de superviser le suivi des maladies et à diriger plus facilement les patients à risque élevé vers un spécialiste.

Dans [5], nous avons montré que la prédiction de l'hospitalisation peut être améliorée en injectant les connaissances métier, qu'elles soient spécialisées ou généralistes dans la représentation vectorielle des dossiers médicaux électroniques.

Nous avons entraîné à partir de données extraites de la base de données PRIMEGE PACA un algorithme d'apprentissage machine capable de prédire l'hospitalisation des patients. Nous avons choisi par conception un algorithme interprétable, car il est crucial pour les médecins de connaître les critères impliqués dans le processus décisionnel de l'algorithme. Des textes ainsi que des analyses biomédicales contenus dans le dossier du patient sont utilisés pour prédire son risque d'hospitalisation et les facteurs impliqués dans cette décision.

La difficulté majeure que l'on a rencontrée était de développer une interface ergonomique qui fournisse toutes les informations nécessaires au médecin généraliste pour l'aider dans sa prise de décision et lui permettre de gagner du temps puisqu'un généraliste sur 5 voit plus de 50 patients par jour (soit le double du nombre recommandé par les directives européennes de sécurité), alors que la moyenne actuelle est supérieure à 40 patients par jour.³ Dans [6], les auteurs ont reporté que des données sur-structurées provoquent la perte de l'attention des médecins, tout comme le fait de devoir naviguer entre de nombreux écrans dans leur système de consultation médicale, ce qui a même pour

2. https://www.has-sante.fr/portail/upload/docs/application/pdf/2015-04/note_methodologique_polyopathie_de_la_personne_agee.pdf

3. <https://collegeofmedicine.org.uk/complementary-medicine-round-up-march-2018/>

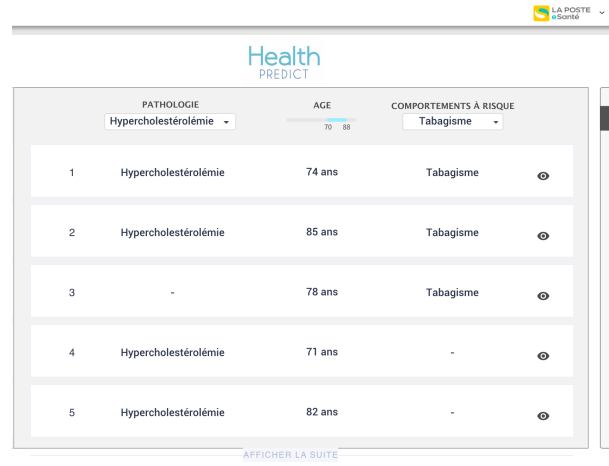


FIGURE 1 – Vue sur le panneau de sélection, existant dans la version prototype, qui permet de sélectionner un patient selon différents critères. Les patients recherchés ici sont atteints d'hypercholestérolémie, âgés de 70 à 88 ans et fumeurs.

effet de les perturber pour identifier des problèmes de santé émergents.

Ainsi, nous avons conçu une interface destinée à des médecins de différentes tranches d'âge, puisqu'en effet, les médecins étant en exercice depuis de nombreuses années sont moins sensibilisés aux nouvelles technologies, et présenté les points essentiels du dossier du patient avec la mise en avant des résultats de l'algorithme de façon visuelle en un nombre réduit d'écrans. De même, pour éviter de surcharger l'interface, les informations complémentaires relatives aux composants graphiques ou le détail sur la fiabilité des méthodes mises en œuvres sont accessibles par le biais de relevés puisqu'une fois connue une aide devient redondante si cette dernière doit s'afficher à chaque utilisation du logiciel.

3 Aperçu du système

3.1 Aperçu de l'interface

L'interface a été déclinée sous deux versions différentes afin d'atteindre des objectifs différents, l'une doit servir de démonstrateur pour montrer les capacités du produit et l'autre doit servir en condition réelle pour aider le médecin généraliste dans son processus décisionnel pour traiter son patient. La principale différence entre les deux versions de l'interface reste la possibilité de sélectionner un patient à partir d'un ensemble de patients (Figure 1) et de lui ajouter une nouvelle analyse biologique ou une nouvelle pathologie, dans l'objectif pour un médecin généraliste de se rapporter à un patient de sa connaissance.

Les facteurs responsables dans la prédiction de l'hospitalisation des patients sont définis sous deux catégories : les facteurs fortement impliqués dans la prévision de l'hospita-

lisation d'un patient et sur lesquels le médecin généraliste peut intervenir, appelés 'Facteurs de risques modifiables' que nous avons représentés dans une pyramide inversée, et ceux qui ont un moindre impact ou sur lesquels le médecin ne peut pas intervenir, appelés 'Autres facteurs' (Figure 4), tous ces facteurs sont classés selon leur importance dans la décision prise par l'algorithme. L'affichage de facteurs ayant un moindre impact peut permettre au médecin généraliste de prendre en compte les différents critères qui figurent dans le dossier d'un patient, en effet, il est possible que le choix d'un traitement soit contre-indiqué pour certaines pathologies ou conditions physiques. Sur le côté sont représentés les pourcentages évalués par l'algorithme d'apprentissage avant la correction d'un facteur de risque et le nouveau pourcentage après la prise en charge d'un problème de santé (Figure 2), afin d'éviter toute confusion quant à la diminution du risque d'hospitalisation, un symbole négatif a été ajouté en face du pourcentage gagné après prise en charge de différents problèmes de santé. En haut de chaque fenêtre, des informations plus génériques sur le patient avec le sexe, l'âge, si le patient a des affections de longue durée (ALD), l'assistance médicale de l'État (AME)⁴ ou la couverture maladie universelle complémentaire (CMU-C)⁵ sont incluses dans un volet (pour ce patient, aucune affection de longue durée, AME ou CMU-C n'est enregistrée).

Sur la vue d'ensemble du dossier du patient, les facteurs de risque sont classés sous différents onglets tels que 'Antécédents', 'Examens biologiques', 'Comportements à risque', d'autres informations complètent la section sommaire du dossier du patient avec 'Allergies / Intolérances' et ses 'Antécédents familiaux'. Sur cet élément (Figure 3), les facteurs déterminés comme modifiables par le médecin sont représentés par un pictogramme symbolisant une loupe afin de signaler qu'il faut s'intéresser de près au problème médical mis en évidence, et seules les analyses en dehors de la norme sont affichées sous l'onglet 'Examens biologiques'. Une spécificité de la version de démonstration est de pouvoir ajouter un nouveau problème de santé et une nouvelle analyse biologique sur cette série d'écrans présentant la vue d'ensemble du dossier du patient, permettant ainsi au médecin généraliste de modifier un cas parmi la présélection de patients qui lui sont proposés, et de se rapporter à un patient qu'il connaît.

3.2 Retour d'expérience

10 médecins généralistes du milieu universitaire ont été consultés au cours d'un groupe de discussion organisé pour obtenir des commentaires sur la conception de l'interface et leurs suggestions sur les éléments à inclure.

Parmi tous les points soulevés⁶, il a été signalé qu'un écran moins alarmiste serait efficace pour rallier le patient à l'avis du médecin généraliste (observance thérapeutique). Ainsi,

4. <https://www.service-public.fr/particuliers/vosdroits/F3079>

5. <https://www.service-public.fr/particuliers/vosdroits/F10027>

6. <https://tinyurl.com/y4w5umue>

Evaluation du risque d'hospitalisation

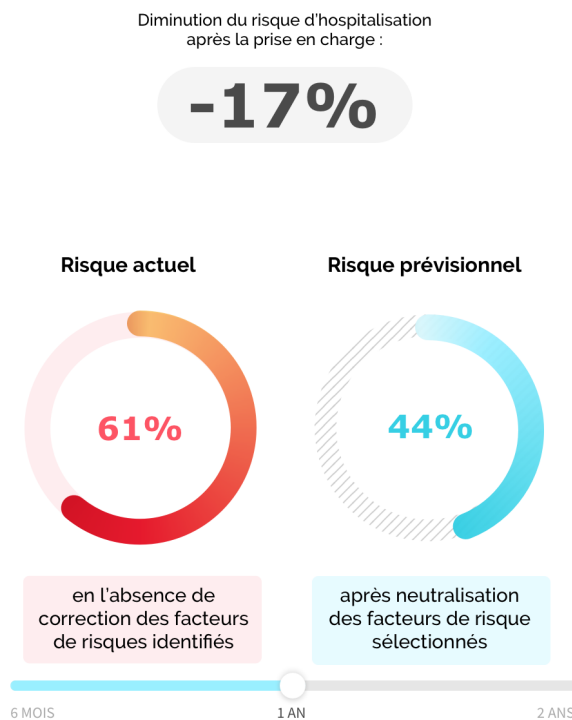


FIGURE 2 – Vue du médecin généraliste avec le risque prévisionnel d'hospitalisation après correction des facteurs 'Tabagisme' et 'Dépression'.

nous proposons de répondre à cette demande par un écran ne montrant que le gain total sur le risque d'hospitalisation représenté dans un demi-cadran, gain calculé en pourcentage relatif. Un bouton en haut à droite de la Figure 2 permet de passer facilement de la vue du médecin à celle du patient. Par défaut nous avons opté pour le calcul du risque d'hospitalisation sur une durée de 1 an, mais un curseur permet de pouvoir ajuster la marge de temps souhaitée pour la prédiction.

Un autre point important soulevé concerne l'importance des explications, en particulier celle relative à un indice de fiabilité sur les résultats obtenus par l'algorithme d'apprentissage. Pour répondre à cette demande, nous avons ajouté plusieurs releveurs qui apparaissent après avoir passé la souris sur l'icône 'i'. L'icône 'i' sur la Figure 2 en particulier, présente des informations sur les méthodes, les métriques utilisées et les publications liées à ce travail, les autres releveurs ont davantage un rôle explicatif sur les éléments de l'interface.

4 Démonstration

Au cours de la session, nous présenterons un poster et une maquette interactive à travers un moniteur, qui montre l'in-

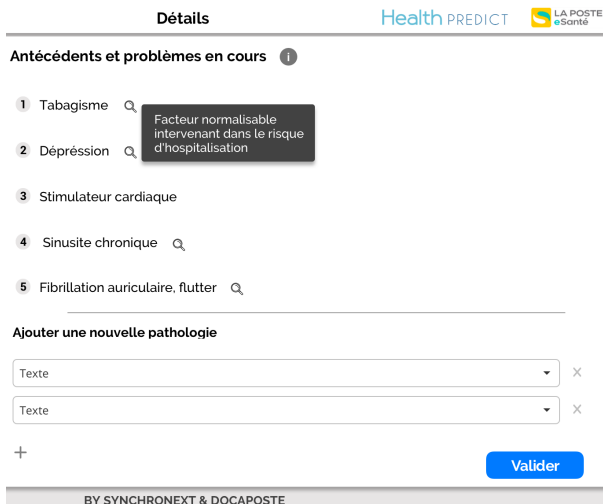


FIGURE 3 – Panneau d’aperçu global sur le dossier du patient, ‘Détails’, sous l’onglet ‘Antécédents’. La version prototype permet d’ajouter de nouvelles pathologies.

terface que nous avons développée sur laquelle un médecin généraliste est capable de réduire significativement le risque d’hospitalisation d’un patient.

Le scénario de notre démonstration comprend les étapes suivantes : (1) nous sélectionnons une pathologie principale, une tranche d’âge ainsi qu’un comportement à risque associés à des patients, (2) dans la liste des patients qui s’affiche nous sélectionnons un patient qui répond aux critères précédemment choisis afin d’obtenir un profil proche d’un patient cible et nous montrons son risque initial d’hospitalisation, (3) nous verrons les différents leviers d’action proposés au médecin généraliste et l’affichage du risque estimé après la prise en charge de certains problèmes de santé du patient, (4) nous nous dirigerons vers l’écran résumant le dossier du patient et nous verrons comment un médecin peut ajouter des pathologies au patient sélectionné dans l’interface de la version prototype dans le but de se rattacher à partir de ce cas à un patient qu’il connaît.

5 Conclusion et travaux futurs

Il est crucial d’explorer, d’analyser et de planifier les facteurs de risque responsables dans l’hospitalisation pour prévenir l’apparition de cet événement et améliorer l’état de santé global du patient. Nous proposons une interface qui répond à ces exigences et qui peut être connectée au produit que nous développons.

L’objectif final de cette application est de se voir directement intégrée à un logiciel de consultation médicale afin d’éviter la double saisie du dossier patient et la multiplication des outils qui menacerait l’efficacité du médecin traitant. La création d’outils aidant le médecin dans sa pratique peut également être une motivation supplémentaire pour lui de remplir avec le plus d’exactitude possible le

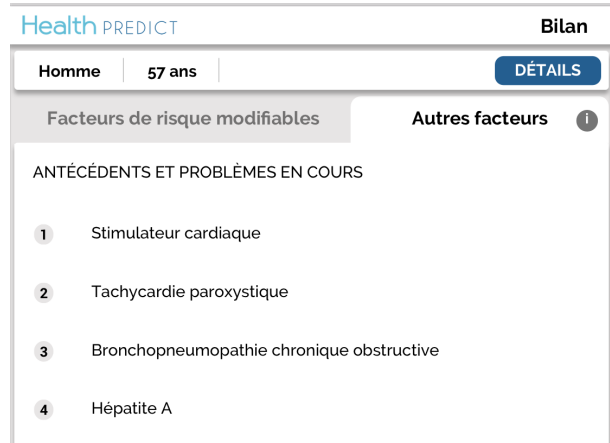


FIGURE 4 – Affichage de l’onglet ‘Autres facteurs’.

dossier de ses patients, ce qui aussi lui permettra d’obtenir des retours sur ses pratiques et celles de ses confrères. Nous prévoyons à court terme d’évaluer et d’adapter cette interface avec les entretiens utilisateurs d’un panel représentatif de médecins généralistes lorsqu’elle sera intégrée à un logiciel de consultation médicale vis-à-vis notamment de la durée passée sur l’interface pour chaque patient ainsi que de la pertinence de la restitution des informations nécessaires au médecin dans la planification d’un traitement, les aspects relatifs à l’impact lié à l’éducation thérapeutique du patient doivent être également étudiés.

Remerciements

Ce travail est en partie financé par le gouvernement français dans le cadre de son projet IDEX UCAJEDI (ANR-15-IDEX-0001).

Références

- [1] V. Lacroix-Hugues, D. Darmon, C. Pradier, and P. Staccini, “Creation of the first french database in primary care using the icpc2 : Feasibility study.,” *Studies in health technology and informatics*, vol. 245, pp. 462–466, 2017.
- [2] C. M. Boyd, J. Darer, C. Boulton, L. P. Fried, L. Boulton, and A. W. Wu, “Clinical practice guidelines and quality of care for older patients with multiple comorbid diseases : implications for pay for performance,” *Jama*, vol. 294, no. 6, pp. 716–724, 2005.
- [3] M. E. Tinetti, S. T. Bogardus Jr, J. V. Agostini, *et al.*, “Potential pitfalls of disease-specific guidelines for patients with multiple conditions,” *N Engl J Med*, vol. 351, no. 27, pp. 2870–4, 2004.
- [4] R. Hillestad, J. Bigelow, A. Bower, F. Girosi, R. Meili, R. Scoville, and R. Taylor, “Can electronic medical record systems transform health care? potential health benefits, savings, and costs,” *Health affairs*, vol. 24, no. 5, pp. 1103–1117, 2005.

- [5] R. Gazzotti, C. F. Zucker, F. Gandon, V. Lacroix-Hugues, and D. Darmon, "Injecting domain knowledge in electronic medical records to improve hospitalization prediction," in *The 16th Extended Semantic Web Conference (ESWC 2019)*, vol. 11503 of *Lecture Notes in Computer Science*, (Cham), pp. 116–130, Springer International Publishing, 2019.
- [6] J. S. Ash, M. Berg, and E. Coiera, "Some unintended consequences of information technology in health care : the nature of patient care information system-related errors," *Journal of the American Medical Informatics Association*, vol. 11, no. 2, pp. 104–112, 2004.

Simultaneous Elicitation of Committee and Voters' Preferences

B. Napolitano¹O. Cailloux¹P. Viappiani²¹ LAMSADE, UMR 7243, CNRS and Université Paris Dauphine, PSL Research University, Paris, France² LIP6, UMR 7606, CNRS and Sorbonne Université, Paris, France

beatrice.napolitano@dauphine.fr, olivier.cailloux@dauphine.fr, paolo.viappiani@lip6.fr

Abstract

Social choice deals with the problem of determining a consensus choice from the preferences of different voters. In the classical setting, the voting rule is fixed beforehand and full information concerning the preferences of the voters is provided. Recently, the assumption of full preference information has been questioned by a number of researchers and several methods for eliciting preferences have been proposed. In this paper we go one step further and we assume that both the voting rule and the voters' preferences are partially specified. In this setting, we present an interactive elicitation protocol based on minimax regret and develop several query strategies that interleave questions to the chair and questions to the voters in order to attempt to acquire the most relevant information in order to quickly converge to optimal or a near-optimal alternative.

Keywords

Computational Social Choice, uncertainty in AI, preference elicitation, minimax regret.

1 Introduction

In a traditional social choice setting, both the social choice function and the full preference orderings of the voters are expressed beforehand. Nevertheless, in real situations this is not always the case. When considering decisions with large sets of alternatives, requiring voters to express full preference orderings can be prohibitively costly and, perhaps, not necessary in order to determine a consensus choice. Furthermore, it is often difficult for non-expert users to formalize a voting rule on the basis of some generic preferences over a desired aggregation method. Consider as an example a situation in which the council of a city wants to build a community allotment and the decision of what to grow has to be taken. The voters are the residents of the neighborhood and the alternatives consist of all the vegetables that can grow in that particular area. An external observer, who is helping with the voting procedure, is responsible for collecting the preferences of both the residents (regarding the vegetable to grow) and the members of the council (regarding the method for aggregating the residents' preferences). Imagine that, after collecting some

preferences, she notices that some alternatives are always the least preferred. In this case, it is really unlikely that these alternatives would be selected as winners by any reasonable voting rule. Thus, she may consider to ask the voters only their preferences over few strategic alternatives instead of asking them the full preference orderings. Furthermore, if she notices that a particular vegetable is the most preferred of almost the totality of the residents there is no need to know the specific voting rule. Any reasonable aggregation procedure, in fact, would select this alternative as a winner. The reasoning behind this is that it is not unrealistic to assume that the committee, that has to decide how to aggregate these preferences, is not able to define a specific procedure.

These observations have motivated a number of recent works considering social choice with partial preference orders [14, 9, 4] and incremental elicitation [3, 8] of voters' preferences. Lu and Boutilier [7] proposed the use of minimax regret to drive incremental vote and preferences elicitation when the social choice function is fixed and known. Furthermore, several authors [12, 6, 13] worked on positional scoring rules with uncertain weights, assuming that the preferences of the voters are fully known. Some elicitation methods for a quite general class of rules based on weak orders have also been proposed by Cailloux and Endriss [2].

In this paper we focus on positional scoring rules, that are a particularly common method used to aggregate rankings, and we assume that both the voters' preferences and the social choice rule are partially specified. We develop methods for computing the minimax-optimal alternative using positional scoring rules and we provide incremental elicitation methods to acquire relevant preference information. We then discuss several heuristics that determine queries, either to a voter or to the committee, that quickly reduce minimax regret. While previous works have considered either partial information about the voters' preferences or a partially specified aggregation method, we do not know of any work considering both sources of uncertainty at the same time.

2 Partial information

We consider a set A of m alternatives (products, restaurants, public projects, etc.) and a set $\{1, \dots, n\}$ of voters. Each voter j comes from an infinite set \mathbb{N} of potential voters and is associated to her “real” preference order \succ_j which is a linear order (connex, transitive, asymmetric relation) over the alternatives. A *profile* is the association of a preference to each voter and it is equivalently represented by $(\succ_1, \dots, \succ_n)$ or by $\mathbf{v} = (v_1, \dots, v_n) \in V$ where $v_j(i) \in \{1, \dots, m\}$ denotes the rank (position) of alternative i in the preference order \succ_j and V is the set of all possible preference profiles. A social choice function associates a profile with a set of winners; we consider *positional scoring rules (PSR)* which attach weights to positions according to the scoring vector (w_1, \dots, w_m) . Without loss of generality, we assume $w_1 = 1$ and $w_m = 0$. An alternative x obtains a score that depends on the rank obtained in each of the preference orders:

$$s^{\mathbf{v}, \mathbf{w}}(x) = \sum_{j=1}^n w_{v_j(x)} = \sum_{r=1}^m \alpha_r^x w_r$$

where α_r^x is the number of times that alternative x was ranked in the r -th position. The winners are the alternatives with highest scores.

In this work we assume fixed, but unknown, a profile $\mathbf{v}^* = (\succ_1^*, \dots, \succ_n^*)$ and a weight vector \mathbf{w}^* . Our knowledge at a given time of the preference of voter j is encoded by a partial order over the alternatives, thus a transitive and asymmetric binary relation, denoted by \succ_j^p ; we assume that preference information is truthful, i.e. $a \succ_j^p b \implies a \succ_j^* b$. An incomplete profile $\mathbf{p} = (\succ_1^p, \dots, \succ_n^p)$ maps each voter to a partial preference. A completion of \succ_j^p is any linear order \succ that extends \succ_j^p and we indicate with $C(\succ_j^p) = \{\succ \in \mathcal{L}(A) \mid \succ_j^p \subseteq \succ\}$ the set of possible completions of \succ_j^p , where $\mathcal{L}(A)$ is the set of linear orders on A .

Our knowledge regarding the weights of the positional scoring rule is represented by a set of constraints restraining the possible values that they can take. We consider a convex sequence of weights and we assume that the chair is able to specify additional preferences about how the social choice function should behave. Such requirements are encoded with linear constraints about the vector \mathbf{w} and the set of these constraints is denoted by \mathcal{C}_W . Given the set of convex weight vectors \mathcal{W} , we use $W \subseteq \mathcal{W}$ to denote the set of weight vectors compatible with the chair's preferences. We will provide an example of such constraints, used to specify information about weights, in Section 4.

3 Robust winner determination

As a decision criterion to determine a winner, we propose to use minimax regret [11]. This concept has been used for robust optimization under data uncertainty [5] as well as in decision-making with uncertain utility values [10, 1].

Lu and Boutilier [7] have adopted this criterion for winner determination in social choice with the preferences of the voters that are only partially known, while the social choice function is predetermined and known. We use *maximum regret* to quantify the worst-case error, which measures, intuitively, how far an alternative is from the optimal one given current knowledge; the alternatives that minimize this error are selected as tied winners. The maximum regret of an alternative is the highest possible difference between its score and the one achievable by any alternative in any state compatible with our current knowledge. It is considered by assuming that an adversary can both 1) extend the partial profile \mathbf{p} into a complete one, and 2) instantiate the weights choosing among any weight vector in W . We formalize the notion of minimax regret in multiple steps. First of all, $\text{Regret}^{\mathbf{v}, \mathbf{w}}(x)$ is the “regret” of selecting x as a winner instead of choosing the optimal alternative under \mathbf{v} and \mathbf{w} :

$$\text{Regret}^{\mathbf{v}, \mathbf{w}}(x) = \max_{y \in A} s^{\mathbf{v}, \mathbf{w}}(y) - s^{\mathbf{v}, \mathbf{w}}(x).$$

The pairwise max regret $\text{PMR}^{\mathbf{p}, W}(x, y)$ of x relative to y given partial profile \mathbf{p} and the set of weights W is the worst-case loss of choosing x instead of y under all possible realizations of the full profile *and* weights. Max regret $\text{MR}^{\mathbf{p}, W}(x)$ is the worst-case loss of x and $\text{MMR}^{\mathbf{p}, W}$ is the value of minimax regret obtained when recommending a minimax optimal alternative.

$$\text{PMR}^{\mathbf{p}, W}(x, y) = \max_{\mathbf{w} \in W} \max_{\mathbf{v} \in C(\mathbf{p})} s^{\mathbf{v}, \mathbf{w}}(y) - s^{\mathbf{v}, \mathbf{w}}(x) \quad (1)$$

$$\text{MR}^{\mathbf{p}, W}(x) = \max_{y \in A} \text{PMR}^{\mathbf{p}, W}(x, y) \quad (2)$$

$$\text{MMR}^{\mathbf{p}, W} = \min_{x \in A} \text{MR}^{\mathbf{p}, W}(x) \quad (3)$$

$$x_{\mathbf{p}, W}^* = \underset{x \in A}{\text{argmin}} \text{MR}^{\mathbf{p}, W}(x) \in A_{\mathbf{p}, W}^* \quad (4)$$

By picking as consensus choice an alternative associated with minimax regret $x_{\mathbf{p}, W}^* \in A_{\mathbf{p}, W}^*$, we can provide a recommendation that gives worst-case guarantees, giving some robustness in face of uncertainty. In cases of ties in minimax regret, we can either decide to return all minimax alternatives $A_{\mathbf{p}, W}^*$ as winners or to pick just one of them using a tie-breaking strategy. Because the constraints in \mathcal{C}_W are not necessarily linear PMR is not straightforwardly optimized, but it can be computed by adapting the reasoning of Lu and Boutilier [7] to the case of uncertain weights using linear programming. This optimization strategy uses the assumption of convexity.

4 Interactive elicitation

We propose an incremental elicitation method based on minimax regret. At each step, the system may ask a question either to one of the voters or to the chair. The goal is to acquire relevant information to reduce minimax regret as quickly as possible. As termination condition of elicitation, we can check whether minimax regret is lower than a

threshold or, if we wish optimality, we can perform elicitation until minimax regret drops to zero.

Question types We distinguish between questions asked to the voters and questions asked to the chair. For the first we consider comparison queries that ask a particular voter to compare two alternatives. Then, the partial profile p is augmented on the basis of the voter's answer. Thus, if voter j answers a comparison query stating that alternative a is preferred to b , then the partial order \succ_j^p is augmented with $a \succ_j^p b$ and by transitive closure. The query for the chair, instead, aims at refining our knowledge about the scoring rule; in particular, we assume we can acquire constraints of the type:

$$w_r - w_{r+1} \geq \lambda(w_{r+1} - w_{r+2})$$

for $r \in \{1, \dots, m-2\}$, relating the difference between the importance of consecutive ranks.

Elicitation strategies A strategy is a function (or a random process) that, given our partial knowledge so far, gives us a question that should be asked.

The *Random* strategy first decides with a probability of $1/2$ each whether it will ask a question about weights or about a preference ordering. Then it equiprobably draws a question among the set of the possible ones. This strategy is clearly non efficient but it gives us a baseline for comparisons.

The *Extreme completions* strategy considers the solution of the minimax regret game, given the current knowledge, and estimates the contribution to the regret of our uncertainty about the weights and the profile. Then, it selects a question for the chair or for the voters depending on which of these two values is the highest.

The *Pessimistic* strategy selects among all the possible questions the one that leads to minimal regret in the worst case. Because all our questions are binary questions, for each of them the pessimistic strategy first assumes the answer is "No", it updates the knowledge according to this negative answer and computes the minimax regret. Then, it does the same assuming the answer is positive. Finally, it uses some aggregation of these two MMR values (for example *max* or *avg*) as a score of the question. The question with the highest score is selected.

The *Two phase* strategy asks a predefined, non adaptive sequence of $m-2$ questions to the chair in order to gather informations about the weights w of the scoring rule, using one question per rank except the extreme ones that we are assuming known ($w_1 = 1$, $w_m = 0$). Then it only asks questions about the voters, using questions as defined in the *Pessimistic* strategy.

5 Conclusions

In this paper we have considered a social choice setting with partial information and we have proposed the use of minimax regret both as a means of robust winner determination as well as a guide to the process of simultaneous elicitation of preferences and voting rule.

The next step is to test our strategies in order to compare the performance of our interleaved elicitation approach that mixes questions to the chair and to the voters, to a more classical approach that elicits the rule first and then the voters' preferences (or vice-versa).

Further development of elicitation strategies, considering alternative heuristics, is also an important direction for future work together with the extension of the approach to voting rules beyond scoring rules.

Acknowledgements: We thank the reviewers for comments helping to improve the paper.

References

- [1] C. Boutilier, R. Patrascu, P. Poupart, and D. Schuurmans. Constraint-based Optimization and Utility Elicitation using the Minimax Decision Criterion. *Artificial Intelligence*, 170(8-9):686-713, 2006.
- [2] O. Cailloux and U. Endriss. Eliciting a suitable voting rule via examples. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, pages 183-188, 2014.
- [3] M. Kalech, S. Kraus, G. A. Kaminka, and C. V. Goldman. Practical voting rules with partial information. *Autonomous Agents and Multi-Agent Systems*, 22(1):151-182, 2011.
- [4] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *IJCAI Workshop on Advances in Preference Handling*, pages 124-129, 2005.
- [5] P. Kouvelis and G. Yu. *Robust Discrete Optimization and Its Applications*. Kluwer, Dordrecht, 1997.
- [6] B. Llamazares and T. Peña. Aggregating preferences rankings with variable weights. *European Journal of Operational Research*, 230(2):348-355, 2013.
- [7] T. Lu and C. Boutilier. Robust approximation and incremental elicitation in voting protocols. In *Proceedings of IJCAI 2011*, pages 287-293, 2011.
- [8] L. Naamani-Dery, I. Golan, M. Kalech, and L. Rokach. Preference elicitation for group decisions using the Borda voting rule. *Group Decision and Negotiation*, 24(6):1015-1033, Nov 2015.
- [9] M. S. Pini, F. Rossi, K. B. Venable, and T. Walsh. Aggregating partially ordered preferences. *J. Log. Comput.*, 19(3):475-502, 2009.
- [10] A. Salo and R. P. Hämäläinen. Preference ratios in multiattribute evaluation (PRIME)-elicitation and decision procedures under incomplete information. *IEEE Trans. on Systems, Man and Cybernetics*, 31(6):533-545, 2001.

- [11] L. J. Savage. *The Foundations of Statistics*. Wiley, New York, 1954.
- [12] W. E. Stein, P. J. Mizzi, and R. C. Pfaffenberger. A stochastic dominance analysis of ranked voting systems with scoring. *European Journal of Operational Research*, 74(1):78 – 85, 1994.
- [13] P. Viappiani. Positional scoring rules with uncertain weights. In *Scalable Uncertainty Management - 12th International Conference, SUM 2018, Milan, Italy, October 3-5, 2018, Proceedings*, pages 306–320, 2018.
- [14] L. Xia and V. Conitzer. Determining possible and necessary winners under common voting rules given partial orders. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*, pages 196–201, 2008.

DeepExpert : vers une Intelligence Artificielle autonome et explicable

R. Orhand^{1,2}

A. Jeannin-Girardon^{1,2}

P. Parrend^{1,3}

P. Collet^{1,2}

¹ Laboratoire Icube - UMR CNRS 7357 – ² Université de Strasbourg – ³ ECAM Strasbourg-Europe
{rorhand, anne.jeannin, pierre.parrend, pierre.collet}@unistra.fr

Résumé

Pour procurer aux intelligences artificielles une meilleure adaptabilité, notre position est de proposer un couplage selon une approche énonciviste et cognitiviste entre réseaux de neurones profonds et systèmes de classeurs. L'autonomie étant un vecteur d'évolution de cette adaptabilité, une définition est proposée dans un premier temps afin d'appuyer dans un second temps les choix et apports des modèles de perception, de raisonnement et de mémoire employés. Ces modèles, que sont respectivement ces réseaux, ces systèmes et la proposition du concept d'arbre d'apprentissage traçant l'expérience du système couplé, doivent permettre à ces intelligences d'identifier elles-mêmes les problèmes issus de leur environnement et concevoir en toute transparence les solutions les plus adéquates.

Mots Clef

Apprentissage Profond, Systèmes Experts, Arbre d'Apprentissage, Autonomie, Enaction, Explicabilité.

Abstract

To provide artificial intelligence with better adaptability, we suggest a coupling based on an enactivist and cognitivist approach between deep neural networks and learning classifier systems. As autonomy is a vector for the development of adaptability, a definition is first proposed to support later the choices and contributions of perception, reasoning and memory models used. Those models, namely these networks, these systems and a learning tree we design, that keeps track of the experiences of the coupled system, should allow these intelligences to identify, by themselves, problems arising from their environment and design in complete transparency the most suitable solutions.

Keywords

Deep Learning, Expert Systems, Learning Tree, Autonomy, Enaction, Explainability.

1 Introduction

L'intelligence artificielle peut être définie par la capacité d'un système à correctement interpréter des données externes, d'apprendre de celles-ci et d'utiliser leurs apprentissages dans le but d'accomplir des tâches [7].

Dans le but de procurer à ces systèmes une meilleure adaptabilité, une piste intéressante est de développer leur

autonomie en s'appuyant sur les fonctions cognitives de l'Homme. Celles-ci sont au moins composées de l'imagination, de la raison et de la mémoire, qu'il est possible de retrouver en partie au sein des ordinateurs, au travers de l'intelligence artificielle [14]. Par ailleurs, deux de ces fonctions cognitives ont été étudiées par Spinoza, selon l'idée que chaque chose se maintienne dans son environnement [15]. Pour ce faire, l'imagination, qui produit des représentations incomplètes, basées sur l'expérience et pouvant être entendue comme la perception, doit servir de support à la raison, afin d'aider ce maintien [9, 15].

Dès lors, une analogie est possible d'une part, entre les réseaux de neurones profonds et la perception et, d'autre part, entre les systèmes de classeurs et la raison : ces réseaux possèdent la capacité d'extraire des représentations intrinsèques aux données et nécessaires à la détection ou classification d'éléments [8], quand ces systèmes sont capables de raisonner à partir de faits en s'appuyant sur une base évolutive de règles et d'apprendre de leurs interactions avec leur environnement [1].

Afin d'étayer cette piste, nous proposons donc, dans ce papier de positionnement, que le concept d'autonomie soit mis en œuvre en couplant, d'une part, des modèles d'apprentissage automatique que sont les réseaux de neurones profonds et, d'autre part, des systèmes experts que sont les systèmes de classeurs. Nous souhaitons de plus donner à ce couplage une mémoire de ses expériences et proposons une nouvelle structure, un arbre d'apprentissage, qui tiendra compte de l'évolution de l'ensemble du système artificiel et lui donnera le droit à l'erreur. Les ordinateurs pourraient ainsi identifier des problèmes et concevoir eux-mêmes les solutions les plus adéquates, tout en exposant le raisonnement élaboré sur les représentations construites.

La notion d'autonomie est d'abord développée en section 2, pour pouvoir expliciter la mise en œuvre du couplage que nous proposons en section 3, avant de conclure dans la section 4.

2 Gagner en autonomie par l'expérience

L'autonomie est définie différemment selon l'interprétation qui est faite de son étymologie grecque ($\alpha\upsilon\tau\acute{o}\varsigma$ et $\nu\acute{o}\mu\omicron\varsigma$ désignant respectivement *soi-même* et *loi*) ou du contexte dans lequel elle est employée. Elle peut renvoyer vers la faculté à se gouverner soi-même selon ses propres règles,

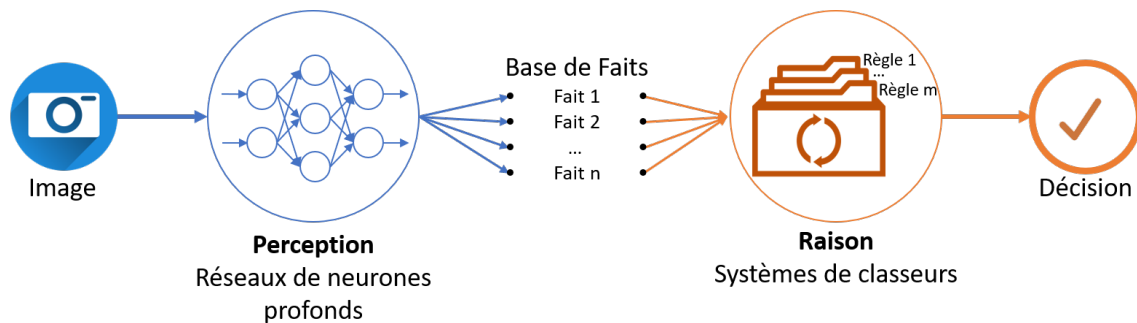


FIGURE 1 – Illustration de DeepExpert : un couplage entre réseaux de neurones profonds et systèmes de classeurs

vers la capacité à être indépendant ou encore vers la génération et le maintien d'une identité [10]. Il n'existe donc pas de définition arrêtée pour l'autonomie et, par extension, de cette faculté pour les ordinateurs.

Cependant, un parallèle est réalisable entre l'autonomie et l'apprentissage automatique, qui désigne la capacité des ordinateurs à apprendre sans être explicitement programmés. Cette définition, attribuée à Samuel [13], souligne le fait qu'un ordinateur se dote de ses propres règles grâce à un apprentissage. Dès lors, nous proposons de définir l'autonomie d'un ordinateur comme sa capacité à réaliser une tâche selon ses propres règles et selon son contexte, en considérant celui-ci du point de vue de l'approche énéactiviste [17], puisque l'ordinateur réalise son apprentissage selon ses interactions avec son contexte.

Le couplage proposé fait ainsi appel à l'apprentissage automatique tant dans son modèle de perception, que dans son modèle de raisonnement, ce qui permet de le faire co-évoluer avec son environnement. De plus, quand les réseaux de neurones font émerger des représentations de l'environnement propres à l'expérience du système artificiel et à ses capacités perceptives, l'usage des systèmes de classeurs rend possible l'interprétation de ses représentations, en générant des règles intelligibles. Il y a donc constitution d'une interface entre l'Homme et le modèle, favorisant ainsi la création de sens en son sein. Nombre de principes de l'énéaction décrits par Varela ou de Loor se retrouvent donc dans ce couplage [4, 17].

Par conséquent, le développement de l'autonomie ne peut s'effectuer sans l'expérience puisqu'elle dépend conjointement de la conception des modèles d'apprentissage, des tâches à réaliser et des interactions avec l'environnement où agit ce modèle. De là, il convient d'explicitier ce système couplant perception, raison et mémoire.

3 Mise en œuvre d'une intelligence artificielle autonome et explicable

L'idée d'une association de réseaux de neurones profonds avec des systèmes de classeurs n'est pas nouvelle : en 2003, Bull proposait les *Neural Classifier Systems* où chaque règle est remplacée par un perceptron multicouche [2]. D'autres modèles ont suivi, tels que les *Neural-Based*

Learning Classifier Systems [3] ou les *Spiking Classifiers* [6], mais un des principaux avantages à l'usage des systèmes de classeurs, qui réside en leur explicabilité grâce à la génération de règles intelligibles, est amoindri dans les modèles cités. Récemment, Matsumoto a proposé un système de classeurs basé sur un auto-encodeur profond, le DCAXCSR [11], dont notre proposition se rapproche. Cependant, l'un des inconvénients majeurs de ce modèle est qu'il ne met pas en œuvre un mécanisme de mémoire tel que recherché dans notre approche cognitiviste et énéactiviste.

Pour pallier ces inconvénients, le modèle issu du couplage proposé entre la perception et la raison, appelé DeepExpert, est illustré figure 1 au travers d'un simple cas d'usage : en prenant l'exemple d'images issues de l'environnement en entrée, les réseaux de neurones profonds les séparent dans des classes distinctes qui constitueront la base de faits, sur laquelle le système de classeur effectuera, selon sa base de règles, des inférences qui aboutiront en une décision (ou action) induisant un changement dans l'environnement perçu par le système.

Concernant la mémoire, il existe différentes manières de la mettre en œuvre dans les systèmes d'apprentissage, telles que les *Long-Short Term Memory (LSTM)* [5]. Les *Memory Networks*, quant à eux, sont des systèmes de mémoire externe permettant de stocker des informations concernant un certain nombre d'états passés du système [16]. Cependant, ces systèmes ont le désavantage d'être figés : on peut alors citer les *Neural Maps* qui contournent cette limitation en autorisant la mise à jour de la mémoire [12].

Néanmoins, dans les *Neural Maps*, l'usage des réseaux convolutionnels profonds et des LSTM impactent l'interprétation des stratégies employées par l'agent : en effet, les techniques d'apprentissage profond employées obscurcissent le raisonnement effectué par l'agent, ce qui rend impossible de justifier avec exactitude les raisons menant à la réalisation de la tâche. Par ailleurs, les *Neural Maps* utilisées sont conçues par l'Homme : l'agent virtuel ne peut donc réaliser des tâches pour lesquelles il n'a pas été développé.

Sur la base de ce modèle, la mémoire de DeepExpert est construite autour de ce que nous appelons un arbre d'apprentissage, dont le but est de tracer l'évolution de l'ap-

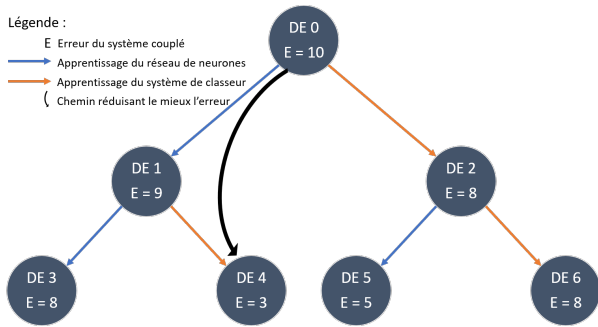


FIGURE 2 – Arbre d'apprentissage de DeepExpert

prentissage de DeepExpert, et de lui donner les outils pour réguler sans intervention humaine son apprentissage.

La figure 2 illustre un tel arbre où chaque nœud représente un état du modèle après apprentissage du réseau de neurones ou du système de classeurs et la racine l'état initial. DE 0 représente donc l'état initial quand le fils gauche DE 1 (respectivement, le fils droit DE 2) représente l'état de DeepExpert après qu'il y ait eu un apprentissage du réseau de neurones (respectivement, du système de classeurs), etc. Comme dans tout processus d'apprentissage, le but est de minimiser l'erreur E du système, erreur qui étiquette chaque nœud. La mise en place d'une telle structure favorise alors le développement de DeepExpert sur le long terme en lui octroyant le droit à l'erreur : sur la figure 2, le nœud DE 1 a une erreur de 9 et le nœud DE 2 a une erreur de 8. Pourtant, l'exploration du nœud DE 1 permet d'obtenir une erreur inférieure à celle qui aurait été obtenue en explorant DE 2.

L'arbre d'apprentissage permet donc à notre système d'avoir une mémoire exploratoire. Elle s'intègre également dans le paradigme énaviviste en permettant par exemple la mise en œuvre du concept d'irréversibilité [4] : pour ce faire, il suffit de définir l'un des nœuds en tant que nouvelle racine et de répéter le processus d'apprentissage. De plus, cette irréversibilité nous permet de contrôler la taille de l'arbre, évitant ainsi l'explosion combinatoire.

4 Conclusion

Pour rendre les intelligences artificielles plus adaptables, l'autonomie est employée en tant que vecteur d'évolution de cette adaptabilité, selon une approche énaviviste et cognitiviste : il s'agit donc de la capacité des ordinateurs à réaliser une tâche selon leurs propres règles et leur contexte, et qui émerge des modélisations de la raison, des perceptions et de la mémoire qui sont des fonctions cognitives de l'Homme. Dès lors, les réseaux de neurones profonds, les systèmes de classeurs et la conception d'un arbre d'apprentissage que nous proposons, représentatifs des perceptions, de la raison et de la mémoire, peuvent être utilisés conjointement pour que ces intelligences puissent identifier elles-mêmes les problèmes issus de leur environnement et concevoir les solutions les plus adéquates.

Les perspectives de ce travail sont nombreuses : en premier lieu, l'utilisation d'un système expert devrait permettre de mieux comprendre les décisions prises par DeepExpert, grâce à la génération de règles interprétables. De plus, l'utilisation des systèmes de classeurs permet d'accroître l'autonomie par la génération de règles par évolution artificielle, permettant ainsi à ce système de pouvoir décomposer de manière autonome une tâche en sous-objectifs, après que le concepteur a spécifié l'objectif global à accomplir. De plus, cette autonomie et cette capacité à générer des sous-objectifs dynamiquement aideraient à mettre en pratique une forme d'apprentissage continu car les nouveaux objectifs permettraient d'ajuster le modèle de perception et ses paramètres, par la nécessité de reconnaître de nouveaux éléments de l'environnement, aidant à la mise en place de nouvelles stratégies pour répondre tant aux sous-objectifs générés, qu'à l'objectif global initial.

Références

- [1] C. Buche, C. Septseault, and P. de Loor, Les systèmes de classeurs. Une présentation générale, *TSI. Techniques et Sciences Informatiques*, vol. 25, no. 8-9, p. 963-990, 2006.
- [2] L. Bull, On Using Constructivism, in *Neural Classifier Systems, Parallel Problem Solving from Nature - PPSN VII*, p. 558-567, 2002.
- [3] H. H. Dam, H. A. Abbass, C. Lokan, and X. Yao, Neural-based learning classifier systems, *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, p. 26-39, 2007.
- [4] P. de Loor, K. Manac'H, and J. Tisseau, Enaction-Based Artificial Intelligence : Toward Co-evolution with Humans in the Loop, *Minds and Machine*, no. 19, p. 319-343, 2009.
- [5] S. Hochreiter and J. Schmidhuber, Long Short-Term Memory, *Neural Computation*, vol. 9, no. 8, p. 1735-1780, 1997.
- [6] D. Howard, L. Bull, and P.L. Lanzi, A cognitive architecture based on a learning classifier system with spiking classifiers, *Neural Processing Letters*, vol. 44, no. 1, p. 125-147, 2016.
- [7] A. Kaplan, M. Haenlein, Siri, Siri, in my hand : Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence, *Business Horizons*, vol. 62, no. 1, p. 15-25, 2019.
- [8] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *Nature*, vol. 521, no. 7553, p. 436-444, 2015.
- [9] F. Lenoir, *Le miracle Spinoza : Une philosophie pour éclairer notre vie*, Fayard, 2017.
- [10] K. Manac'H, *Vers la notion d'agent énaviv virtuel : Application à l'approche dynamique évolutionnaire*, Université Européenne de Bretagne, 2011.
- [11] K. Matsumoto, R. Takano, T. Tatsumi, H. Sato, T. Kovacs, and K. Takadama, XCSR based on compressed input by deep neural network for high dimensional data, *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, p. 1418-1425, 2018.
- [12] E. Parisotto, and R. Salakhutdinov, Neural map : Structured memory for deep reinforcement learning, arXiv :1702.08360, 2017.
- [13] A. L. Samuel, Some Studies in Machine Learning Using the Game of Checkers, *IBM Journal of Research and Development*, vol. 3, no. 3, p. 210-229, 1959.
- [14] M. Serres, Les nouvelles technologies : révolution culturelle et cognitive, *Interstices*, INRIA, 2007.
- [15] B. Spinoza, *Ethique*, III, prop. 6.
- [16] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, End-to-end memory networks, *Advances in neural information processing systems*, p. 2440-2448, 2015.
- [17] F. J. Varela, *Invitation aux sciences cognitives*, Paris, Seuil, 1988.

Planification dans les jeux-vidéos : À quoi servent les coûts des actions ? *

Guillaume Prévost^{1,2}

Stéphane Cardon¹

Tristan Cazenave²

Éric Jacopin¹

¹ CREC Saint-Cyr, Écoles de Coëtquidan, F-56381 GUER

² LAMSADE - Université Paris-Dauphine, F-75775 Paris Cedex 16

guillaume.prevost@st-cyr.terre-net.defense.gouv.fr

Résumé

Nous discutons de l'intérêt des coûts des actions dans le planificateur G.O.A.P. (Goal-Oriented Action Planning), qui est utilisé pour le contrôle en temps-réel des personnages non-joueurs dans certains jeux-vidéos. Nous avons instrumenté l'activité de planification du jeu vidéo F.E.A.R. afin d'étudier l'usage des coûts des actions par le planificateur. Nous montrons et nous expliquons pourquoi leur impact est faible en pratique.

Mots Clef

Planification d'actions, A*, Coût des actions, Jeux-vidéo

Abstract

This paper suggests a reflection on the costs of the actions used by the planner G.O.A.P. (Goal-Oriented Action Planning) which is used in some video games like F.E.A.R. to have real-time control of the non-playable character. We have analysed the planning activity of F.E.A.R. to study the utility of the costs in the planner. We show and explain why they have little impact in practice.

Keywords

Actions planning, A*, Actions cost, Video Game.

1 Introduction

La planification d'actions en temps-réel pour la conception de jeux vidéo a pour avantage de découpler les actions des différents états possibles d'un personnage non-joueur (PNJ). La conception des actions est ainsi facilitée car leur ajout ou leur retrait ne nécessitent plus de modifier l'intégralité du système. Le couplage se fera en jeu, par le planificateur, qui générera des plans en chainant les actions candidates. Le système G.O.A.P. (Goal-Oriented Action Planning), créé en 2003 [5] et mis en œuvre pour la première fois en 2005 dans le jeu vidéo F.E.A.R. par Jeff Orkin [6], est un algorithme de planification basé sur celui de STRIPS [2]. Il est utilisé dans de nombreux jeux et notamment dans "Shadow of Mordor" [4] ou encore "Rise

of the Tomb Raider" [1]. Il fonctionne de la manière suivante : les buts d'un PNJ représentent des états souhaités du monde (états buts); les différents états du monde sont modélisés par des symboles; chacun des PNJs possède des capteurs (vision, vie, sons...) et une mémoire afin de percevoir et de comprendre le milieu où ils se trouvent; les actions possèdent un coût et sont équipées de *préconditions* et de *postconditions* (ou *effets*) qui ne sont, ni plus ni moins, que des symboles d'état du monde; enfin, la génération de plans est faite grâce à l'algorithme de plus court chemin A* [3], qui est déjà mis en œuvre pour la recherche de trajectoire pour le déplacement des PNJs. Nous avons fait finir le jeu F.E.A.R. à trois personnes différentes et au moins deux fois. L'une des deux parties était configurée avec tous les coûts d'action mis à 1 tandis que la deuxième se déroulait avec les coûts d'origine. L'étude des données expérimentales récoltées durant ces sessions montre que l'ensemble des plans d'actions générés dans la configuration des coûts originaux (CO) sans l'être dans celui des coûts unitaires (CU) est faible à 5,14%. Ainsi, dans G.O.A.P., à quoi servent les coûts des actions? Le premier argument avancé est qu'il permet de forcer un choix dans le planificateur. Entre autres, dans F.E.A.R., plus une action sera spécifique plus son coût sera faible afin de la favoriser. Par exemple, les actions *Attack* et *AttackFromCover* ont le même effet mais leur coût diffère avec une valeur de 6.0 pour la première et de 4.0 pour la deuxième. Il y a 18 actions différentes qui possèdent pour seul et unique effet celui d'*Attack* et d'*AttackFromCover* et leur coût varie de 1.0 à 7.0 (Certaines ont le même coût). C'est la catégorie la plus variée. L'autre raison pour l'existence des coûts est que cela permet la réutilisation de A* qui est mis en œuvre pour le déplacement des PNJs. Pour ce faire, l'algorithme de plus court chemin a besoin d'une fonction heuristique et d'une fonction de coûts pour sommer les deux résultats afin de choisir le meilleur candidat à la frontière du graphe de recherche. Soit "n" un état du monde intermédiaire entre l'état initial et l'état but, sachant que la recherche de plans commence depuis l'état but pour atteindre l'état initial, la fonction heuristique évalue le nombre de symbole d'état du monde à satisfaire entre "n" et l'état ini-

*Les travaux décrits sont financés par SAFRAN dans le cadre de la chaire « Soldat Augmenté ».

tial tandis que la fonction de coût est la somme du coût des actions nécessaires pour atteindre "n" depuis l'état but. Par exemple, imaginons un PNJ qui n'a pas son arme chargée et qui reçoit le but *KillEnemy*. Le symbole de l'état du monde *TargetIsDead* doit avoir la valeur *true*. La fonction coût de l'état but est nulle car aucune action n'est choisie pour l'instant tandis que sa fonction heuristique vaut 1.0 puisqu'un symbole est non-satisfait. Les attaques *Attack* et *AttackMelee* sont proposées au planificateur puisque leur effet satisfait le symbole d'état du monde. Les préconditions de *AttackMelee* sont toutes satisfaites alors qu'*Attack* a besoin du symbole *WeaponLoaded* mis à *true*. *AttackMelee* qui a un coût de 3.0 pointe sur un état du monde qui n'a pas de symbole insatisfait, il s'agit de l'état initial, donc la fonction heuristique vaut 0.0 et sa fonction d'évaluation vaut 3.0 (3.0 + 0.0). *Attack* qui a un coût de 6.0 pointe sur un état du monde qui a un symbole insatisfait, donc sa fonction heuristique vaut 1.0 et sa fonction d'évaluation vaut 7.0. Imaginons que cette dernière action soit choisie (Ce qui n'arrivera pas en pratique car $3.0 < 7.0$), l'action *Reload* sera proposée, elle a un coût de 5.0 et pointe sur l'état initial. L'état pointé a donc une fonction heuristique qui vaut 0.0, une fonction de coût qui vaut 11.0 (6.0 + 5.0) et, finalement, une fonction d'évaluation qui vaut 11.0. La suite du papier est organisée de la façon suivante : un premier paragraphe présentant tous les résultats obtenus avec les figures fournies en annexe et un second paragraphe exposant les discussions et les conclusions.

2 Résultats

Sur les six sessions (une session représentant le jeu en entier) avec les deux configurations, 270.866 plans ont été générés : 139.939 plans pour les coûts originaux(CO) avec 19h59 de temps de jeu et 130.927 plans pour les coûts unitaires(CU) avec 18h18 de temps de jeu. Parmi ces plans, 83 sont différents dans la configuration CO et 68 dans la configuration CU. 65 sont en commun et ils représentent 94,86% des plans pour CO et 99,96% des plans pour CU. Les plans générés avec la configuration CU sont donc inclus dans ceux de la configuration CO. La diversité engendrée par les coûts originaux représente 5,14%. La Figure 2 montre que la probabilité d'apparition d'un plan, en fonction du nombre d'actions le composant, est la même, que les coûts soient originaux ou que les coûts soient unitaires. Les échecs représentent respectivement 74,05% et 76,88% des plans générés. Un échec correspond à l'incapacité du planificateur à retourner au moins un plan valide pour résoudre un but. C'est le cas lorsque qu'aucune action n'est candidate ou lorsqu'au moins une précondition de chaque action candidate ne peut être satisfaite. Ensuite, 103.626 plans sont réussis pour CO : 68,32% possèdent une seule action, 30,50% possèdent deux actions, 0,86% en possèdent trois et 0,32% en possèdent quatre. Pour CU, 130.927 plans sont réussis : 66,45% n'ont qu'une seule action, 32,49% en ont deux, 0,79% en ont trois et 0,27% en ont quatre. La majeure partie des plans n'ont donc qu'une

ou deux actions puisqu'ils représentent 98,9% des plans générés réussis dans les deux configurations. La Figure 1, qui compare la probabilité d'apparition des 58 plans en commun entre les deux configurations, nous permet d'observer que l'influence des coûts des actions est majoritairement négligeable sur le comportement du planificateur de F.E.A.R.. Les plans d'actions ont une probabilité d'apparition équivalente pour chaque configuration à plus ou moins 0.5% de différence. Une différence plus prononcée est néanmoins remarquable sur le plan : *{Attack}*. Concernant le temps de construction d'un plan, le changement de coûts n'a aucun impact. Dans les deux cas le planificateur met un temps de l'ordre de la dizaine de micro seconde pour proposer un plan. Cette analyse est observable sur la Figure 3, qui présente les diagrammes à moustaches des temps d'exécution des plans en fonction de la configuration et de leur taille.

3 Discussion et conclusion

Dans la configuration actuelle de G.O.A.P. pour le jeu F.E.A.R., les coûts des actions sont très peu déterminants pour le contrôle en temps réel des PNJs : ils n'influencent ni sur le temps de construction ni sur le comportement des PNJs. Bien que les coûts des actions soient un paramètre essentiel pour la conception du planificateur utilisant l'algorithme de recherche A*, ils ne sont pas impactant car, d'une part les plans sont courts (98.9% n'ont au plus que deux actions) donc ils ne permettent pas d'exploiter le potentiel d'A*, et d'autre part le système G.O.A.P. dans F.E.A.R. utilise deux méthodes d'élagage particulièrement efficaces qui permettent de sélectionner des actions candidates précises. La première sélection se fait à l'aide des postconditions, le planificateur ne garde que les actions qui possèdent le bon effet. Ensuite, une méthode d'élagage utilisant la mémoire et les capteurs du PNJ est implémentée dans chaque classe d'action. Cette méthode permet de ne garder que les actions qui, parmi toutes celles qui ont le bon effet, sont acceptables dans le contexte de la situation. Ainsi, si un PNJ n'est pas au contact du joueur mais est en mesure de tirer, l'action *Attack* pourra être candidate tandis que l'action *AttackMelee* sera ignorée bien que les deux actions aient le même effet : "*TargetIsDead = true*". Cette méthode d'élagage contextuelle supplante le rôle des coûts en réduisant le facteur de branchement de A*. Elle explique l'impact si peu important présenté Figure 1. La probabilité d'apparition plus prononcée pour le plan *{Attack}* dans la configuration CU ou la diversité de 5,14% de la configuration CO est due au peu de fois où A* a eu plusieurs propositions et où la configuration du coût en fonction de la spécificité de l'action a eu, pour CO, ou aurait eu, pour CU, un intérêt.

Références

- [1] Chris Conway. Goap in tomb raider. GDC AI Summit, 2015.

- [2] Richard Fikes and Nils Nilsson. STRIPS : A new approach to the application of theorem proving to problem solving. *AI Journal* 2(5), pages 189–208, 1971.
- [3] Peter Hart, Nils Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on System Science and Cybernetics*, pages 100–107, 1968.
- [4] Peter Higley. Goal-oriented action planning : Ten years old and no fear ! GDC AI Summit, 2015.
- [5] Jeff Orkin. *Applying Goal-Oriented Action Planning to Games*. 2003.
- [6] Jeff Orkin. Agent architecture considerations for real-time planning in games. *Proceedings of the 1st AIIDE*, pages 105–110, 2005.

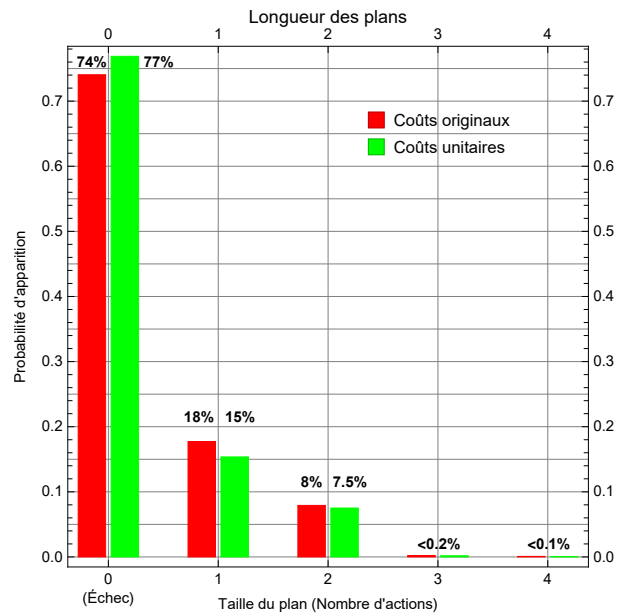


FIGURE 2 – Probabilité d’apparition des plans en fonction de leur longueur (0 étant un échec de planification).

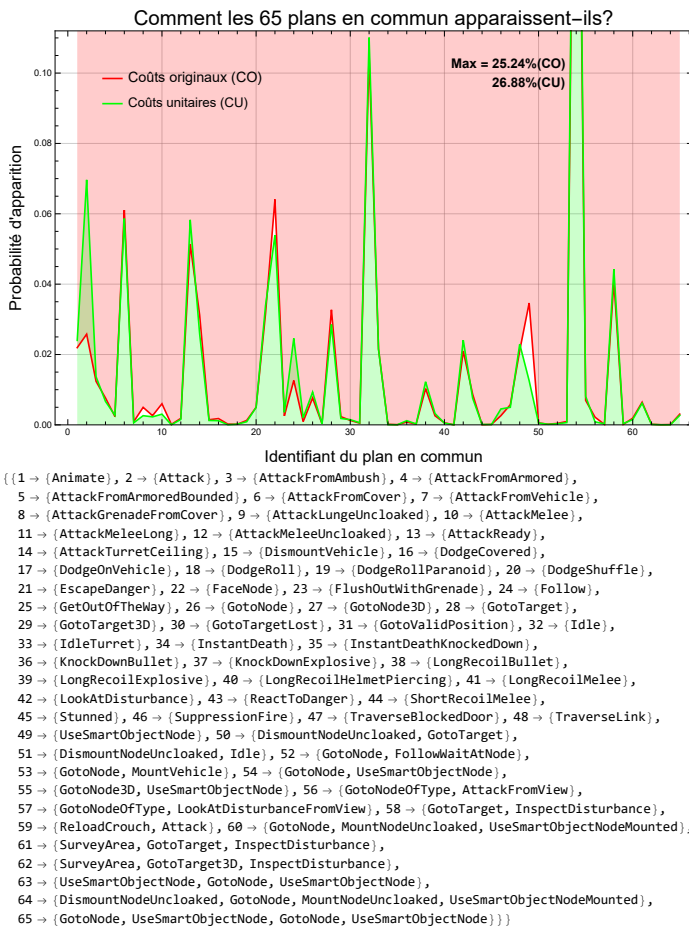


FIGURE 1 – Comparaison des probabilités d’apparition des plans en commun entre les coûts originaux et les coûts unitaires.

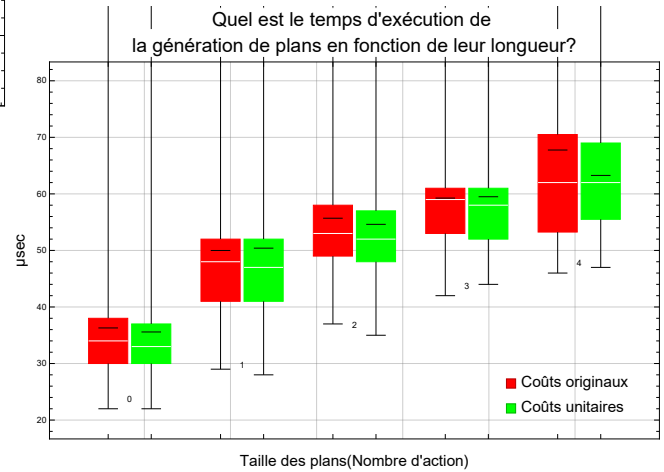


FIGURE 3 – Diagrammes à moustache des temps d’exécution du planificateur; la marque blanche dénote la valeur médiane tandis que la marque noire dénote la valeur moyenne.

Diagnostic de l'atrophie multi-systématisée par l'utilisation de réseaux de neurones convolutifs multi-modaux à 3 dimensions

Edouard Villain¹ & Federico Nemmi², Anne Pavy-Le Taron^{3,4}, Olivier Rascol⁵, Xavier Franceries⁶,
Patrice Péran² & Marie-Véronique Le Lann¹

¹ LAAS-CNRS, Université de Toulouse, CNRS, INSA, UPS, Toulouse, France

² ToNIC, Toulouse NeuroImaging Center, Université de Toulouse, Inserm, UPS, France

³ Institut des Maladies Métaboliques et Cardiovasculaire, Inserm, UMR 1048, Toulouse, France

⁴ Department of Neurology and Institute for Neurosciences, University Hospital of Toulouse, France

⁵ Departments of Clinical Pharmacology and Neurosciences, Clinical Investigation Center CIC 1436, NS-Park/FCRIN network and NeuroToul COEN Center, Inserm, CHU de Toulouse, Université de Toulouse, France

⁶ Inserm CRCT, UMR 1037, Toulouse, France

evillain@laas.fr

Résumé

L'atrophie multi-systématisée (AMS) est une maladie neurodégénérative. Le diagnostic et le suivi des patients atteints d'AMS est un des objectifs de la neuroimagerie moderne. L'imagerie par résonance magnétique multi-modale (IRMm) a montré des résultats prometteurs. Les dernières avancées en matière de réseaux de neurones convolutifs ont été utilisées pour créer une architecture sur la base des données IRMm capable de discriminer les patients atteints d'AMS de sujets sains. Les résultats ont montré (i) l'efficacité des réseaux de neurones pour discriminer patients AMS et sujets (ii) la possibilité d'extraire une signature spatiale.

Mots Clef

Atrophie multi-systématisée, Réseaux de neurones, N-échantillons validation croisée, Syndrome Parkinsonien.

Abstract

Multiple system atrophy (MSA) is a neurodegenerative disease. The diagnosis and follow-up of patients with AMS is one of the objectives of modern neuroimaging. Multi-modal magnetic resonance imaging (mMRI) has shown promising results. The latest advances in convolutional neural networks have been used to create an architecture based on mMRI data that can discriminate between patients with MSA and healthy subjects. The results showed (i) the effectiveness of neural networks in discriminating MSA patients and subjects (ii), the ability to extract a spatial signature.

Keywords

Multiple system atrophy, Neural network, N-fold cross validation, Parkinson's syndrome.

1 Introduction

L'atrophie multi-systématisée (AMS) est une maladie neurodégénérative. Le diagnostic et le suivi des patients atteints d'AMS est un des objectifs de la neuroimagerie moderne. L'imagerie par résonance magnétique multi-modale (IRMm) a montré des résultats prometteurs [1, 2].

Les récentes avancées en matière d'intelligence artificielle et plus particulièrement en réseaux de neurones (RdN) permettent de plus en plus d'applications dans le domaine médical [3]. Nous tenterons de répondre à deux questions : (i) l'approche par RdN permet-elle de discriminer patients AMS et sujets sains lorsque les données sont de l'ordre de quelques dizaines (ii) l'approche RdN nous permet-elle d'extraire une signature spatiale multiparamétrique discriminante.

1.1 Travaux antérieurs

Le neurone artificiel s'appuie sur l'analogie avec le neurone biologique. Les transmissions d'information effectuées biologiquement via les dendrites et les axones correspondent respectivement aux entrées $[x_1, x_2, \dots, x_n]$ et à la sortie y du neurone artificiel. Le traitement de l'information effectué par le corps cellulaire est quant à lui assuré par le calcul $\varphi\{\sum_1^n(x_i\omega_i) + b\}$ avec ω_i les poids du neurone, b son biais et φ sa fonction d'activation. L'apprentissage du neurone artificiel consiste à déterminer ses poids et le biais de façon à minimiser l'erreur de prédiction.

Le réseau de neurones est ensuite créé en connectant la sortie d'un neurone aux entrées d'autres neurones. De nombreuses architectures de réseaux de neurones sont proposées dans la littérature et permettent de répondre à différentes tâches [4, 5].

Le réseau de neurones convolutif utilise les opérations de convolutions et de sous-échantillonnages afin d'apprendre des formes sur le signal d'entrée à différentes échelles. L'architecture est alors organisée en couches et se termine généralement par des couches de neurones entièrement connectées.

2 Jeu de données

Un jeu de données annoté et relativement équilibré de 55 patients (29 AMS contre 26 sujets sains) a été utilisé dans cette étude et correspond à un sous ensemble des données analysées dans [1, 2]. Pour chacun des patients, trois modalités pertinentes pour le diagnostic des troubles liés au cerveau ont été utilisées :

- le volume de matière grise (GM) qui rend compte du volume où l'activité électrique intracérébrale a lieu et qui a tendance à se réduire dans certaines pathologies cérébrales neurodégénératives.
 - la diffusivité moyenne (MD) qui permet d'imager le mouvement Brownien des particules d'Hydrogènes. Les patients atteints de syndrome Parkinsonien présentent des agrégats ferreux qui vont modifier ce mouvement Brownien.
 - l'amplitude de fluctuation des basses fréquences (ALFF) qui mesure l'activité cérébrale au repos par l'IRM fonctionnelle en imageant l'oxygénation des globules rouges.
- Chaque modalité est un volume isotrope 3 dimensions de 60x72x60 voxels, dont chaque voxel est de résolution physique 3x3x3mm³.

3 Méthodologie

Nous avons développé une architecture générique de réseau de neurones convolutif multi-modale 3D schématisée en Fig. 1, de façon à pouvoir analyser les résultats obtenus pour chaque combinaison des modalités dont nous disposons. Le cœur de l'architecture s'appuie sur l'enchaînement de deux couches de convolutions suivies d'une couche de regroupement [6].

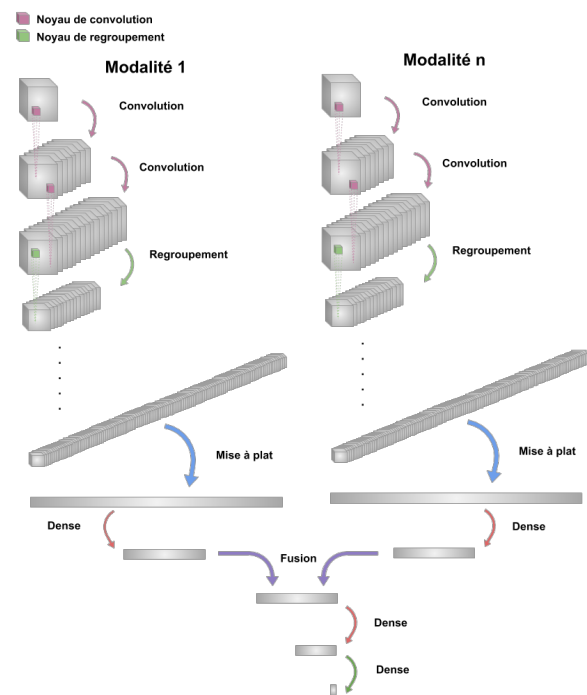


Figure 1 : Architecture du réseau de neurones convolutif 3D multi modale

Une validation croisée à 10 plis, répétée 10 fois, a été calculée pour chacun de nos modèles, ce qui permet d'analyser leur robustesse vis à vis des itérations tout en utilisant un jeu de données de taille restreinte [7].

Une exploration des activations des modèles est ensuite opérée dans le but de détecter les zones anatomiques pertinentes pour le modèle dans sa décision.

4 Résultats et Discussion

L'analyse des probabilités des prédictions moyennes des modèles nous indique que la modalité ALFF a une probabilité moyenne des prédictions inférieure à 0,9, tandis que la modalité MD semble plus pertinente avec 0,975 en moyenne. Tous les modèles utilisant cette modalité MD ont une probabilité moyenne des vrais positifs et vrais négatifs supérieure à 0,97 comme le montre la Fig. 2, avec :

- les vrais positifs (VP) définis comme les patients AMS ayant obtenus un test positif,
- les vrais négatifs (VN) définis comme les sujets sains ayant obtenus un test négatif,
- les faux positifs (FP) définis comme les sujets sains ayant obtenus un test positif,
- les faux négatifs (FN) définis comme les patients MSA ayant obtenus un test négatif.

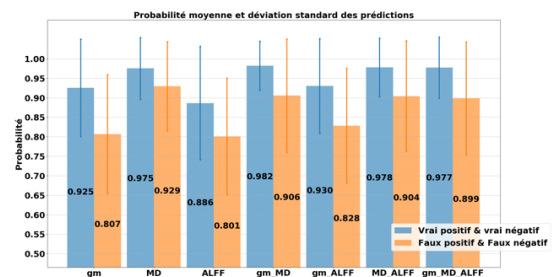


Figure 2 : Probabilité et déviation standard des vrais positifs et négatifs (bleu) ainsi que des faux positifs et négatifs (orange) par modèle

La sensibilité représente la probabilité que le test soit positif si le patient est atteint de l'AMS, mesurée chez les patient AMS seulement et se calcule via l'équation (1).

$$\text{sensibilité} = \frac{VP}{VP + FN} \quad (1)$$

La spécificité quant à elle représente la probabilité que le test soit négatif si le sujet est sain, mesurée chez les sujets sains seulement et se calcule via l'équation (2).

$$\text{spécificité} = \frac{VN}{VN + FP} \quad (2)$$

L'exactitude est alors la probabilité qu'un test soit correct, i.e. test positif pour un patient AMS et test négatif pour un sujet sain, mesurée sur l'ensemble de la population et se calcule via l'équation (3).

$$\text{exactitude} = \frac{VP + VN}{VP + VN + FP + FN} \quad (3)$$

Le tableau 2 détaille les résultats obtenus pour les modèles MD et la combinaison GM_MD_ALFF.

Le modèle utilisant la modalité MD seule fournit un meilleur score d'exactitude, une meilleure stabilité vis-à-vis des itérations et ce dans un temps dix fois plus court. Le modèle utilisant la combinaison des trois modalités obtient quant à lui un score de spécificité similaire, mais avec un score de sensibilité plus faible.

	MD	GM_MD_ALFF
Exactitude	86.2 ± 1.6 %	82.5 ± 2.7 %
Sensibilité	85.2 ± 1.7 %	79.3 ± 4.0 %
Spécificité	87.3 ± 4.8 %	86.2 ± 1.9 %
Durée d'apprentissage	2 minutes	25 minutes

Tableau 2 : Résultats obtenus pour le modèle utilisant la modalité MD et le modèle multi-modal (trois modalités)

Ensuite l'analyse des localisations spatiales des activations qu'utilise le modèle MD a été réalisée dans le but de repérer les zones anatomiques pertinentes pour le diagnostic de l'AMS. Pour cela, les données d'entrées de patients sont utilisées dans un réseau de neurones déjà entraîné, puis la totalité des résultats intermédiaires obtenus sur les couches de convolutions pour chacun des filtres, ainsi que sur les couches de normalisation est sauvegardée. L'une des problématiques est la quantité de données récupérées à analyser. C'est pourquoi, nous nous sommes orientés vers les activations maximales et la Fig. 3 nous présente celles obtenues pour la modalité MD.

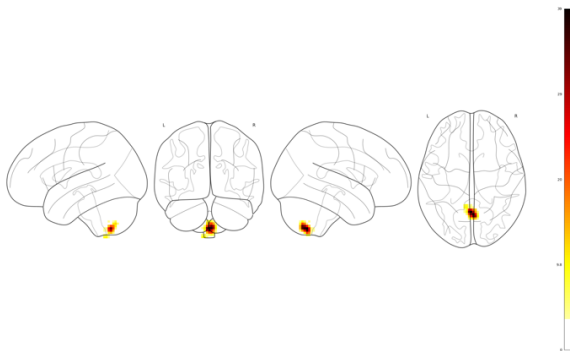


Figure 3 : Activations maximales de la diffusivité moyenne

5 Conclusion

Nous avons développé une architecture de réseau de neurones convolutif 3D multi-modal permettant de discriminer patients AMS et sujets sains avec des scores d'exactitude de l'ordre de 85 %.

L'étude des activations des modèles permet de repérer les zones pertinentes dans leurs décisions et, elles sont en adéquation avec les résultats obtenus précédemment [1, 2] : la diffusivité moyenne est la modalité la plus pertinente pour le diagnostic de l'AMS et plus précisément dans le cervelet et le putamen.

Bibliographie

- [1] P. Péran *et al.*, MRI supervised and unsupervised classification of Parkinson's disease and multiple system atrophy: MRI Classification in Parkinsonian Syndromes, *Movement Disorders*, 2018.
- [2] F. Nemmi *et al.*, A totally data-driven whole-brain multimodal pipeline for the discrimination of Parkinson's disease, multiple system atrophy and healthy control, *NeuroImage*, 2019.
- [3] L. Zou *et al.*, 3D CNN Based Automatic Diagnosis of Attention Deficit Hyperactivity Disorder Using Functional and Structural MRI, *IEEE Access*, 2017.

- [4] M. R. Mohammadi *et al.*, A brief review over neural network modeling techniques, *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017.
- [5] W. Liu *et al.*, A survey of deep neural network architecture and their applications, *Neurocomputing*, 2017.
- [6] K. Simonyan & A. Zisserman *et al.*, Very Deep Convolutional Networks for Large-Scale Image Recognition, *CoRR*, 2014.
- [7] J. Rodríguez *et al.*, Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation, *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 2010.

Démonstration de l'assistant de viabilité Viduc pour la gestion participative dans un éco-socio-système

Laetitia Zaleski¹

¹ Sorbonne Université, CNRS, Laboratoire d'Informatique de Paris 6, LIP6, Paris

zaleski@poleia.lip6.fr

Résumé

Cette note présente le projet de démonstration de l'assistant Viaduc. La démonstration montrera comment l'assistant permet d'analyser la viabilité des propositions d'un utilisateur à partir de ses objectifs et de ses moyens d'action. L'assistant vise à un soutien informatisé à la prise de décisions en matière de gestion collective participative des aires protégées. Ce projet consiste en un assistant pour la négociation permettant aux utilisateurs la formulation d'un projet cohérent, compatible avec ceux des autres utilisateurs, et facilitant la recherche de compromis. Nous proposons d'utiliser les concepts et outils de la théorie mathématique de la viabilité et nous illustrons notre approche avec un modèle de parc marin.

Mots-clefs

Viabilité, Aide à la décision, Gestion participative.

1 Introduction

Le tourisme de faune peut apporter des avantages financiers aux communautés locales tout en contribuant à la préservation de la biodiversité, en augmentant significativement le commerce régional (Ballantyne et Packer) [1]. La gestion des aires protégées est un problème récurrent dans le domaine de la préservation de la biodiversité [2]. En effet, ces zones comptent de nombreuses ressources, ce qui peut entraîner des conflits entre les différents acteurs sociaux (ONG de protection des tortues, association de pêcheurs, opérateur de tourisme...). Nous reprenons l'analyse de Briot et al. [3] sur les différents acteurs et types de gestion, à laquelle nous ajoutons un assistant pour la négociation grâce à l'utilisation de la théorie de la viabilité [4], de manière individuelle et collective [5]. Nous nous inspirons d'un parc marin situé à Arraial do Cabo, dans l'État de Rio de Janeiro au Brésil, où touristes, tortues et pêcheurs interagissent. Sur la figure 1 on donne l'exemple d'un tel parc où pêcheurs,



FIGURE 1 – Sur cette image, on observe que touristes, pêcheurs et tortues marines se partagent un même espace, chacun influençant les autres.

touristes et tortues se partagent un même espace. Les pêcheurs sont des pêcheurs traditionnels qui vivent de la pêche et proposent des services aux touristes ce qui leur permet de recevoir un complément de revenu. Ils prennent accidentellement des tortues dans leurs filets. Les touristes viennent au parc principalement pour observer les tortues marines, et dégradent leur environnement. Ces interactions sont expliquées plus en détails dans la suite.

Notre approche s'inspire en partie de celle du collectif ComMod (Companion Modeling) [6] qui vise à accompagner les populations locales par l'intermédiaire de modélisation multi-agent. Cependant, notre parc n'est pas simulé à l'aide de systèmes multi-agents mais par un système d'équations, permettant l'utilisation de la théorie de la viabilité. Notre modèle s'inspire de ce parc marin mais n'est pas calibré sur ce parc. Il nous permet la construction d'un assistant à la négociation, permettant des analyses comparatives de différents points de vue (en fonction des objectifs et des contraintes fixés par les différents utilisateurs), afin de faciliter une base de négociation non ambiguë.

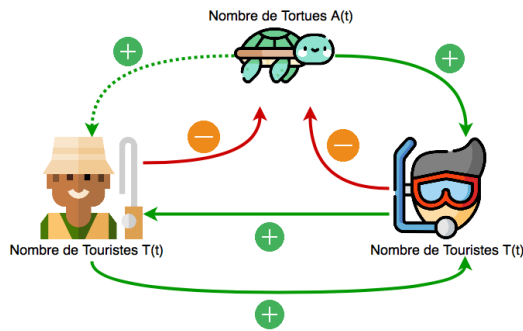


FIGURE 2 – Influence des variables les unes sur les autres : les touristes (T) et le capital des pêcheurs (C) ont un impact négatif sur le nombre de tortues (A) (flèche signée "-"), mais le nombre de tortues et le capital des pêcheurs sont attractifs pour les touristes (flèche signée "+"). Le capital des infrastructures est indirectement influencé par le nombre de tortues car nous considérons que ce nombre est proportionnel au nombre de poissons, il s'agit de la flèche pointillée.

2 L'assistant

Viaduc, notre assistant Viabilité, s'inscrit dans le contexte d'un jeu sérieux pour la gestion de parcs protégés appelé Simparc [3]. A ce jeu, nous ajoutons une dimension dynamique et nous nous intéressons aux apports de la viabilité dans le cadre de la gestion participative d'aires protégées, ce qui n'a encore jamais été étudié à notre connaissance. A ce jour, l'assistant est implémenté dans un mode mono-utilisateur avec ses fonctionnalités de bases décrites par la suite. Dans le futur, nous souhaitons que notre assistant propose une aide plus poussée et soit étendu à plusieurs utilisateurs en parallèle.

2.1 Modélisation du problème

Afin d'utiliser la viabilité, l'assistant nécessite un modèle, qui peut dépendre des différents scénarios. A l'heure actuelle nous avons un scénario de parc marin et notre modèle prend en compte la population animale (de tortues) $A(t)$, l'activité touristique $T(t)$ et le Capital des infrastructures de pêche $C(t)$, où t représente le temps dans les trois cas. Les touristes se rendent au parc marin pour (entre autres) observer ces tortues et la zone protégée est ouverte à la pêche. Les pêcheurs offrent des services aux touristes (excursions en bateau par exemple). Leur pêche conduit à la capture accidentelle de tortues marines. Pour définir le modèle, nous nous sommes inspirés du travail sur le tourisme durable de Casagrandi et Rinaldi [7]. Comme décrit dans la figure 2. Nous choisissons ici un modèle simple, avec un nombre de variables réduit, afin que les calculs de viabilité décrit par la suite se fassent dans un temps raisonnable.

Des actions de contrôle sont possibles et font partie intégrante de la modélisation. Nous avons choisi pour ce modèle deux actions de contrôle : la modulation de la proportion du parc ouverte au tourisme et l'effort de restauration environnementale. Ces actions ne sont pas fixes et peuvent varier dans une certaine plage.

2.2 La viabilité

La théorie mathématique de la viabilité étudie la compatibilité entre un système dynamique et un ensemble de contraintes. Dans le problème que nous considérons, les acteurs sociaux définissent les contraintes acceptables et les moyens d'action. Nous utilisons dans la suite la théorie de la viabilité, introduite par Aubin en 1991 [4]. La théorie de la viabilité nous permet de considérer un système dynamique influencé par des actions de contrôle. Ces actions peuvent varier dans le temps afin de maintenir le système dans un ensemble de contraintes. Chaque participant peut décider quels sont les états souhaitables, en fonction de ses préférences. En utilisant la théorie de la viabilité, l'assistant est capable de montrer aux utilisateurs s'il est possible de maintenir le système dans un état désiré en utilisant les actions de contrôle qu'ils ont suggéré. Plutôt que de définir des critères pour l'optimalité du système (par exemple, maximiser le nombre de tortues dans le parc), nous voulons considérer les contraintes sur les valeurs possibles des différentes variables du modèle, qui devraient être respectées pour que le système soit et demeure viable. La viabilité a déjà fait ses preuves en matière de développement durable [8, 9], dans cet article, la viabilité nous sert également d'outil à la gestion participative et à la négociation.

La noyau de viabilité. La théorie de la viabilité considère un système dynamique défini par son état $\vec{x} \in X \subset \mathbb{R}^n$, dont l'évolution peut être influencée par un contrôle \vec{u} tel que :

$$\begin{cases} \dot{\vec{x}}(t) = f(\vec{x}(t), \vec{u}(t)) \\ \vec{u}(t) \in U(\vec{x}(t)) \end{cases}$$

où $U(\vec{x}(t)) \subset \mathbb{R}^q$ ($q \in \mathbb{N}$) est l'ensemble des contrôles admissibles et $\vec{u}(t)$ est une fonction de contrôle pour f .

Le noyau de viabilité est défini comme l'ensemble des états du système pour lesquels il existe toujours (au moins) une fonction de contrôle qui maintient le système dans les limites des contraintes, et est défini comme suit :

$$\text{Viab}(K) = \{\vec{x} \in K \mid \exists \vec{u}(\cdot) \forall t \in [0, +\infty], \vec{x}^*(t) \in K\}$$

L'ensemble K peut être défini par un système classique de contraintes statiques (par exemple $K = (x, y) \mid a \leq x \leq b, c \leq y \leq d$) La figure 3 montre un exemple de noyau de viabilité : pour tous les points de la zone en violet, il existe au moins une séquence de contrôle qui maintient la dynamique dans l'état souhaité.

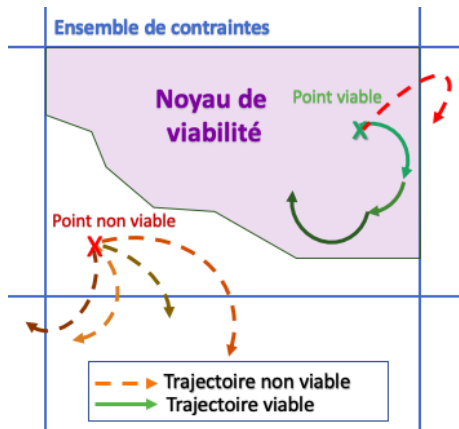


FIGURE 3 – Exemple d'un noyau de viabilité. La zone en violet correspond au noyau de viabilité. Les lignes pointillées sont les trajectoires non viables et les lignes pleines sont les trajectoires viables. Les lignes de différentes couleurs représentent des politiques de contrôles différentes. Plusieurs algorithmes permettent de calculer un noyau de viabilité [10, 11]

2.3 Le mode mono-utilisateur

L'assistant mono-utilisateur propose deux types d'aide. En premier lieu, l'aide à la cohérence du projet personnel. L'assistant permet à une personne de se rendre compte si ses moyens d'actions (représenté ici par des contrôles) et ses contraintes sont compatibles avec le modèle en calculant un noyau de viabilité. Ce mode mono-utilisateur permet aussi dans un second temps de préparer l'utilisateur à la négociation qui sera faite dans le mode multi-utilisateur. La figure 4 regroupe des captures d'écran de l'assistant, qui est actuellement sous la forme d'une application web où l'utilisateur peut choisir ses contraintes ainsi que ses actions de contrôles et l'assistant peut ainsi calculer le noyau de viabilité correspondant et apporter différentes information sur le noyau.

Le cas du noyau vide : L'assistant peut être utilisé en mode mono-utilisateur afin de déterminer si le projet personnel de l'utilisateur est cohérent. Le projet est cohérent s'il est possible d'atteindre l'objectif de l'utilisateur (matérialisé par des limites inférieures et supérieures sur les trois variables Figure 4b) étant données les hypothèses du modèle en utilisant les paramètres de contrôle (Figure 4c) choisis par ce dernier. Si les contraintes souhaitées ne sont pas compatibles avec la dynamique, ou si les intervalles de contrôle sont trop faibles, l'assistant calculera un noyau de viabilité vide. Cela signifie que l'aire protégée n'est pas viable sous les contraintes choisies. Dans ce cas, l'utilisateur peut assouplir ses contraintes en en proposant un ensemble plus large ou rendre son action plus efficace en ajoutant

plus de possibilités de contrôle. L'assistant de viabilité peut aider les utilisateurs dans cette tâche en proposant de nouveaux ensembles de contrôles et de contraintes. Actuellement l'assistant propose le noyau correspondant au jeu de paramètres le plus proche parmi ceux stockés dans la base de données. Notre base de donnée contient à ce jour des noyaux calculés lors des précédentes utilisations de l'assistant. Dans le futur nous prévoyons une exploration systématique de l'espace des paramètres (contraintes et contrôles) à partir de suites de type Sobol [12] qui s'ajoutera au stockage des noyaux déjà calculés à la demande des utilisateurs. Les noyaux les plus proches sont déterminés à l'aide d'une distance euclidienne dans un espace à 10 dimensions (trois variables et deux paramètres de contrôles, possédant chacun une borne minimum et maximum).

2.4 Travaux futurs

Le cas du noyau non vide : Si l'utilisateur a trouvé un noyau correspondant aux contraintes, l'assistant peut offrir de multiples informations sur ce noyau, et par extension, sur la difficulté à rester dans ce noyau. Par exemple, il est possible de vérifier ce qu'il se passe si l'on interdit aux trajectoires du modèle de s'approcher de la frontière du noyau. En effet, une fois que le système arrive près de la frontière du noyau, il devient plus difficile à contrôler. Il est possible de calculer un noyau érodé, c'est à dire un ensemble compris dans le noyau de viabilité, et pour lequel on interdit aux trajectoires de passer près de la frontière du noyau. Si le volume du nouveau noyau érodé est trop faible par rapport au noyau précédent, cela suggère un manque de robustesse du noyau par rapport à la définition du problème de viabilité.

L'assistant est pensé pour pouvoir être utilisé en mode multi-utilisateur, permettant aux participants de partager des informations sur leurs contraintes et leurs noyaux. Dans ce mode, l'assistant pourra donner de multiples conseils pour la négociation future entre les participants. Le but étant de savoir s'il est possible de trouver une politique de contrôle sur laquelle tous les participants sont d'accord.

3 Conclusion

Dans cet article, nous avons proposé d'appliquer la théorie de la viabilité pour l'aide à la gestion. Bien que l'assistant s'appuie sur notre modèle de parc marin, inspiré d'une réserve naturelle brésilienne, il est pensé pour pouvoir être utilisé avec d'autres modèles sans que les étapes ne changent. La viabilité permet aux utilisateurs d'avoir une base commune et non ambiguë pour la négociation. A l'heure actuelle un prototype mono-utilisateur a été implémenté et testé sans joueurs.

Welcome to Viaduc:

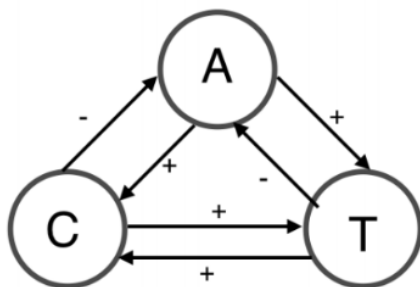
What is Viaduc ?

Viaduc is a Viability Expert Agent based on a viability analysis (Wei et al., 2012). It is based on the Viability theory described by Aubin (1992) During a game session, the player is able to choose constraints, controls (i.e how to control the system in order to keep it within the constraints) and will be able to change the value of some parameter if he or she doesn't agree with the dynamic.capacity of the viability expert agent to help a player to analyze one viability kernel corresponding to a set of constraints that he himself decided, but also to compare with alternative kernels/constraints explored by himself or proposed by other players during the negotiation. Therefore, this provides the players with a basic way to quantify and analyze the degree of feasibility and viability of proposals. Instead of just comparing the constraint sets, the viability expert compares the viability kernels, which are based on the link between the dynamics and the constraints. Small changes in constraint sets can have a broad range of impacts depending on the dynamics.

About the equations

C represents the fishermen's capital (boats, income...), A represents the number of Animals (turtles) in the parc, T represents the number of tourists in the parc

Scheme



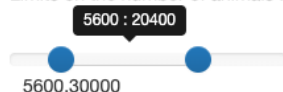
Equation

Step 1: Set up the parameters:

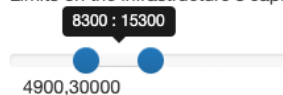
(a) Après une courte présentation de l'assistant, l'utilisateur se familiarise avec le modèle. Ensuite il a la possibilité de modifier les paramètres, d'observer la dynamique, puis de choisir ses contraintes ainsi que ses moyens de contrôles.

Choose the limits on A, C and T:

Limits on the number of animals :



Limits on the infrastructure's capital :



Limits on the number of tourist :



(b) L'utilisateur doit choisir une limite inférieure et supérieure sur les trois variables du modèle.

Step 1 :Choose the controls:

Use the control epsilon

Epsilon represents the effort to restore the environment, 100 is a maximum effort and 0 is no effort at all

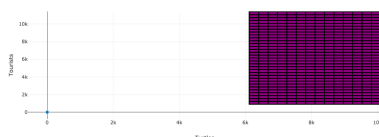


Use the control zeta

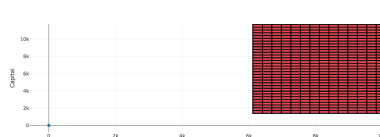
Zeta represents the areas reachable by the tourists, at 0.2 we consider that the parc is completely open, tourism



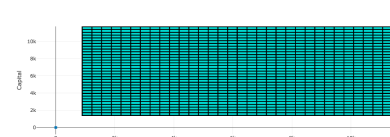
(c) Après avoir choisit une limite inférieure et supérieure sur les trois variables du modèle l'utilisateur choisit ses moyens d'actions



(d) Calcul et affichage de projections des trois faces du noyau par l'assistant : Projection de la face Touristes / Tortues



(e) Projection de la face Capital des infrastructures de pêche / Tortues



(f) Projection de la face Capital des infrastructures de pêche / Touristes

FIGURE 4 – Captures d'écran de l'assistant de viabilité

Références

- [1] R. Ballantyne and J. Packer, *International handbook on ecotourism*. Edward Elgar Publishing, 2013.
- [2] C. Trave, J. Brunnschweiler, M. Sheaves, A. Diedrich, and A. Barnett, "Are we killing them with kindness? Evaluation of sustainable marine wildlife tourism," *Biological conservation*, vol. 209, pp. 211–222, 2017.
- [3] J.-P. Briot, A. Sordoni, E. Vasconcelos, V. S. Patto, D. Adamatti, M. Irving, and C. Lucena, "Design of an artificial decision maker for a human-based social simulation-experience of the simparc project," *Activity-based modeling and simulation*, pp. 17–35, 2010.
- [4] J.-P. Aubin and H. Frankowska, "Viability kernel of control systems," in *Nonlinear synthesis*, pp. 12–33, Springer, 1991.
- [5] L. Zaleski, I. Alvarez, J.-P. Briot, and M. de Azevedo Irving, "Viaduc : Un assistant de viabilité pour la négociation dans un éco-socio-système," *RJCIA*, July 2019.
- [6] C. ComMod, "La modélisation comme outil d'accompagnement," *Natures Sciences Sociétés*, vol. 13, no. 2, pp. 165–168, 2005.
- [7] R. Casagrandi and S. Rinaldi, "A theoretical approach to tourism sustainability," *Conservation ecology*, vol. 6, no. 1, 2002.
- [8] C. Bernard, *La théorie de la viabilité au service de la modélisation mathématique du développement durable. Application au cas de la forêt humide de Madagascar*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II, 2011.
- [9] A. Beall and L. Zeoli, "Participatory modeling of endangered wildlife systems : Simulating the sage-grouse and land use in central washington," *Ecological Economics*, vol. 68, no. 1-2, pp. 24–33, 2008.
- [10] P. Saint-Pierre, "Approximation of the viability kernel," *Applied Mathematics and Optimization*, vol. 29, no. 2, pp. 187–209, 1994.
- [11] I. Alvarez, R. Reuillon, and R. De Aldama, "Viabilitree : A kd-tree framework for viability-based decision," 2016.
- [12] A. Saltelli, *Global Sensitivity Analysis*. Wiley, 2008.

