



HAL
open science

Actes des 18es Rencontres des Jeunes Chercheurs en Intelligence Artificielle

Alexandre Niveau

► **To cite this version:**

Alexandre Niveau. Actes des 18es Rencontres des Jeunes Chercheurs en Intelligence Artificielle : RJCIA 2020. Plate-Forme Intelligence Artificielle, Association Française pour l'Intelligence Artificielle, 2020. hal-04566976

HAL Id: hal-04566976

<https://ut3-toulouseinp.hal.science/hal-04566976>

Submitted on 2 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 Chercheur·ses
en Intelligence Artificielle*

PFIA 2020

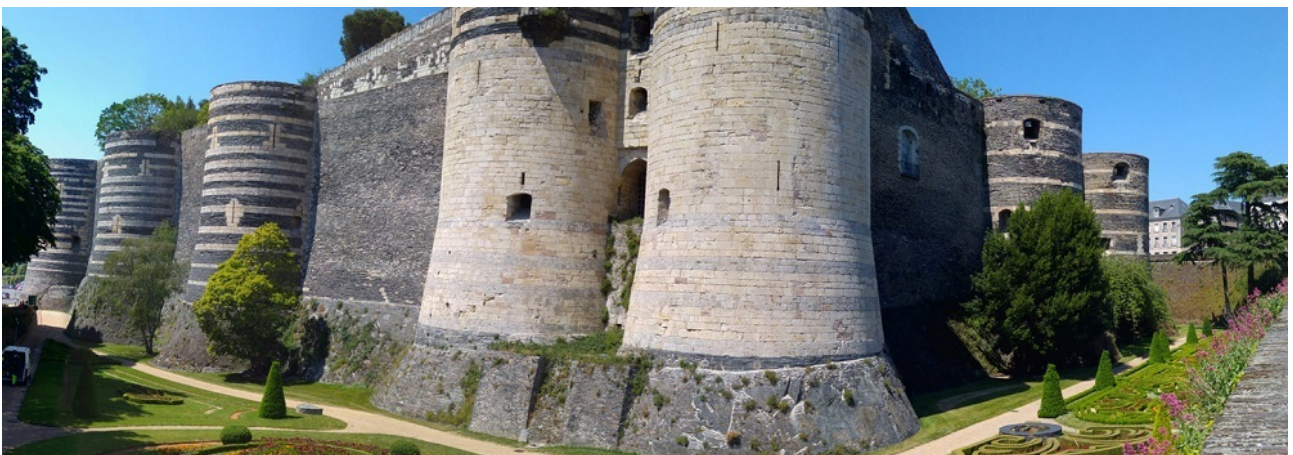


Table des matières

| | |
|---|----|
| Alexandre NIVEAU | |
| Éditorial | 5 |
| Comité de programme | 6 |
| Jialin LIU (conférence invitée) | |
| What evolutionary computation can do for games? | 7 |
| Détection d'objets et raisonnement spatial | |
| Axel MASCARO, Christophe REY | |
| Recommandation ontologique multicritère pour la métrologie | 8 |
| Hiba ALQASIR, Damien MUSELET, Christophe DUCOTTET | |
| Double adaptation de domaine pour la détection d'objets | 17 |
| Jérémy CHOPIN, Jean-Baptiste FASQUEL, Harold MOUCHÈRE, Isabelle BLOCH, Rozenn DAHYOT | |
| Méthode d'analyse sémantique d'images combinant apprentissage profond et relations structurales par appariement de graphes | 24 |
| Hippolyte DUBOIS, Patrick LE CALLET, Antoine COUTROT | |
| Génération automatique de graphe pour modéliser les singularités de trajectoires spatiales ... | 32 |
| Théorie des jeux et recherche opérationnelle | |
| Josselin GUÉNERON, Grégory BONNET | |
| Un modèle de jeux de coalitions stochastiques répétés | 34 |
| Paul COURTIN, Axel GRIMAULT, Mehdi LHOMMEAU, Jean-Baptiste FASQUEL | |
| Problème d'affectation dynamique des emplacements de stockage chez Knapp : vers de l'apprentissage automatique? | 42 |
| Ons NEFLA, Meltem OZTURK, Paolo VIAPPANI, Imène BRIGUI-CHTIOUI, Oussama RABOUN | |
| Group-based ordinal classification based on a negotiation process | 50 |
| Algorithmes décentralisés | |
| Iris LOHJA, Yves DEMAZEAU, Christine VERDIER | |
| A multi-agent system approach to ridesharing for older people – State-of-the-art work and preliminary design | 53 |
| Alaa DAOU, Flavien BALBO, Paolo GIANESSI, Gauthier PICARD | |
| Approche décentralisée d'insertion avec amélioration continue de la qualité de la solution pour un système TAD | 61 |
| Maxime HOUSSIN, Stéphanie COMBETTES, Marie-Pierre GLEIZES, Berangere LARTIGUE | |
| SANDMAN : un système auto-adaptatif pour la détection d'anomalies dans le flux des données des bâtiments intelligents | 69 |
| Apprentissage profond | |
| Gaëlle JOUIS, Harold MOUCHÈRE, Fabien PICAROUGNE, Alexandre HARDOUIN | |
| Tour d'horizon autour de l'explicabilité des modèles profonds | 77 |
| Tristan GOMEZ, Harold MOUCHÈRE, Thomas FRÉOUR, Magalie FEYEU | |
| Apprentissage profond appliqué à la classification d'images microscopiques embryonnaires ... | 84 |
| Louis RIGAL, Jeremy CHOPIN, Jean-Baptiste FASQUEL, Raphaël FERNANDEZ, Corinne CHALLE-BLANLOEIL | |
| Apprentissage profond et localisation d'objets : deux exemples d'applications en agriculture .. | 93 |

Aide à la décision et interfaces humain-machine

Valentin FOUILLARD, Safouan TAHA, Nicolas SABOURET, Frédéric BOULANGER

Diagnostiquer les biais cognitifs 100

IA et assistance à la décision humaine : influence des informations transmises sur l'évaluation d'un nouveau plan de vol.

Adrien Metge, Nicolas Maille 108

Aurélien LAMERCERIE

Transduction sémantique pour la modélisation de système 115

Mohamed EL FATRI, Francis ROUSSEAUX, Pierre DREUX

Travail du futur : des portails e-collaboratifs intrinsèquement motivants 122

Agents autonomes

Edison CHUNG, Maxime LEFRANÇOIS, Olivier BOISSIER

Accroître l'interopérabilité dans le Web des objets à l'aide d'agents autonomes (prise de position)
126

Thomas ALVES, Jeremy RIVIERE, Cedric ALLAUX, Yves LE COMTE, Frank SINGHOFF, Thierry DUVAL, Vincent RODIN

Modèle d'interruption pour la répartition des tâches : application pour une simulation de colonie d'abeilles..... 132

Éditorial

Les Rencontres des jeunes chercheuses et chercheurs en Intelligence artificielle sont destinées aux doctorant·es ou titulaires d'un doctorat depuis moins d'un an. L'objectif de cette manifestation est double :

- permettre aux jeunes qui débudent la recherche en intelligence artificielle de se rencontrer et de présenter leurs travaux, et d'ainsi former des contacts avec d'autres jeunes chercheur·ses et d'élargir leurs perspectives en échangeant avec des spécialistes d'autres domaines de l'intelligence artificielle ;
- former les jeunes chercheur·ses 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 chercheur·ses de leur domaine ou de domaines connexes.

Avec 26 articles soumis, et 19 acceptés (dont 3 courts), cette édition des RJCIA a connu un succès dont on ne peut que se réjouir. Il est d'autant plus regrettable que la conférence n'ait pu avoir lieu que virtuellement !. . . Néanmoins, cela n'a impacté qu'une partie des missions de la conférence — en effet, le processus de soumission et de relecture a pu avoir lieu relativement normalement, donnant lieu à des rapports de relecture se voulant bienveillants et constructifs. Ne reste qu'à espérer que la virtualisation n'ait pas trop restreint les discussions, qui sont l'autre aspect important de ces « *rencontres* » !

Alexandre NIVEAU

Comité de programme

Président

- Alexandre Niveau, Université de Caen Normandie

Membres

- Céline Alec, Université de Caen Normandie
- Zied Bouraoui, Université d'Artois
- Mathieu Chollet, IMT Atlantique
- Quentin Cohen-Solal, Université Paris Dauphine
- Nicolas Cointe, Delft University of Technology (Pays-Bas)
- Jérôme Delobelle, Inria
- Géraud Fokou, Inria
- Hugo Gilbert, Gran Sasso Science Institute (L'Aquila, Italie)
- Pierre-François Gimenez, LAAS
- Juan Antonio Lossio-Ventura, Stanford University (États-Unis)
- Jean-Guy Mailly, Université Paris Descartes
- Abdelkader Ouali, Université de Caen Normandie
- Anastasia Paparrizou, Université d'Artois
- Éric Piette, University of Maastricht (Pays-Bas)
- Abdallah Saffidine, University of New South Wales (Australie)
- Michael Sioutis, Aalto University (Finland)
- Marta Soare, Université d'Orléans
- Michaël Thomazo, Inria
- Nicolas Verstaevel, Université Toulouse I Capitole
- Anaëlle Wilczynski, Technische Universität München (Allemagne)

What evolutionary computation can do for games ?

Conférence invitée de Jialin LIU

Abstract This talk introduces the motivation of applying evolutionary computation to games and illustrates how evolutionary computation techniques have been applied to board games and video games using some examples, including the use of evolutionary computation for learning policies, planning game action sequences (e.g., rolling horizon), and designing (e.g., search-based procedural content generation, the combination of evolutionary computation and machine learning methods for procedural content generation).

Recommandation ontologique multicritère pour la métrologie

Axel Mascaro, Christophe Rey

Université Clermont Auvergne, LIMOS

{axel.mascaro,christophe.rey}@uca.fr

Résumé

La recommandation et le classement d'informations sont des processus d'aide à l'utilisateur qui lui propose la ou les meilleures réponses à sa requête. En l'absence d'une réponse exacte, lui offrir les propositions les plus proches possibles de son besoin est une amélioration de ce processus. Dans le contexte d'une plateforme de recherche d'information métrologique, nous discuterons du problème de classement par l'intermédiaire de la représentation des connaissances, avec une approche basée sur les Logiques de Descriptions, les ontologies et la comparaison multicritère. Nous proposerons un raisonnement permettant de comparer entre elles plusieurs propositions, à l'aide notamment de la différence et la découpe en plusieurs composantes d'une information. Nous présenterons une heuristique mettant en application ces résultats.

Mots-clés

Recommandation, classement, logique de description, Ontologie, comparaison multicritère.

Abstract

Matchmaking and information ranking are processes to help users, by offering them the best answers possible at their request. When there is no exact answer, giving them the closest proposition available is an efficient upgrade of that helping process. With a research platform on metrology as a framework, we will discuss about ranking with knowledge representation, with an approach based on Description Logic, ontologies and multicriteria comparison. We present a reasoning to compare each proposition with the other, with semantic and syntactic difference, by truncating the information in distinct component.

Keywords

Matchmaking, ranking, Description Logic, Ontology, multicriteria comparison.

1 Introduction

STAM¹ est un projet d'une plateforme multi-usage dans le domaine de la métrologie. Elle permettra à n'importe quel industriel de standardiser une description d'une mesure, la comparer avec d'autres existantes et obtenir des valeurs et une incertitude révisée en fonction de son contexte

de mesure. Le projet cherche aussi à créer une communauté autour de la métrologie et de fournir une plateforme de documentation sur le sujet. C'est sur cette dernière partie que ce travail porte.

La plateforme en cours propose déjà un système de recherche de ressources. Une ressource représente une information ou un fichier stocké sur la plateforme. Elle est identifiée par des caractéristiques définies dans une ontologie de la métrologie et qui sont utilisées lors des recherches. Les facettes sont des annotations qui font partie de ces caractéristiques et elles sont communes à plusieurs ressources. Par la sélection d'une facette, l'utilisateur va pouvoir restreindre les résultats de sa recherche aux ressources annotées par la facette choisie. Par exemple, la facette "Instrument" permet de trier les ressources obtenues en réponse à une recherche en fonction des instruments de mesure évoqués dans les ressources. Il est possible pour l'utilisateur de choisir plusieurs facettes et d'utiliser la recherche par mots-clés en même temps.

Nous proposons de compléter cette recherche par un processus de recommandation de ressources basée sur une recherche approximative par rapport à la sémantique de leurs descriptions. On cherche alors à offrir, en plus des résultats exacts lors de recherche par facette, les résultats approximatifs les plus pertinents : les ressources les plus proches sémantiquement des facettes et mots-clés de la requête. Nous proposons ainsi d'envisager la recherche approximative de ressources comme un raisonnement de plus proche mise en correspondance sémantique (matchmaking sémantique) de descriptions de ressources avec une requête utilisateur, basé sur une ontologie du domaine de la métrologie et piloté par une approche de comparaison multicritère.

En section 2 nous faisons des rappels concernant les logiques de description, formalisme de représentation des connaissances et de raisonnement utilisé ici pour définir la proximité sémantique d'une ressource avec une requête. A la section 3, nous proposons notre raisonnement. Plus précisément, en section 3.1, nous définissons la notion de composante nécessaire à la comparaison des requêtes et ressources. Puis, en section 3.2, nous formalisons la notion de classement sémantique dans cadre des logiques de description permettant le tri des ressources au niveau de chaque composante. En section 3.3, nous proposons une première heuristique multicritère synthétisant le classement sémantique sur l'ensemble des composantes. En section 4, nous évoquons quelques travaux existants de matchmaking sé-

1. https://limos.fr/news_project/118

mantique et nous situons notre approche par rapport à eux. Enfin, nous concluons en section 5.

2 Logiques de descriptions

Les logiques de description (LD) est un formalisme de représentation des connaissances et de raisonnement sur celles-ci. Plus précisément, elles constituent une famille de sous-langages de la logique du premier ordre munies de la même sémantique basée sur la théorie des modèles, mais aussi d'une syntaxe particulière basée sur les notions de concept, de rôle et d'individu. On considère que l'on a un ensemble d'éléments appelés domaine. Un concept est identifié par un nom comme *Instrument* ou *Material*. Un concept définit un ensemble d'éléments du domaine, comme le ferait une classe en langage objet. Un rôle est identifié par un nom comme *hasMaterial* ou *hasInstrument*. Un rôle définit un ensemble de couples d'éléments du domaine. Par exemple, le rôle *hasMaterial* relie ici un instrument avec son matériau constituant. Il définit un ensemble de couples d'éléments dont le premier est un instrument et le second est le matériau constituant cet instrument. Un individu est identifié par un nom comme *RegleBois2* ou *Acier1*. C est un élément du domaine.

Plus formellement, la sémantique des connaissances décrites par les LD est définie par la notion d'interprétation. Une interprétation est une paire $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, où $\Delta^{\mathcal{I}}$ est le domaine qui regroupe l'ensemble des individus étudiés et $\cdot^{\mathcal{I}}$ la fonction d'interprétation qui relie tous les concepts à une partie de $\Delta^{\mathcal{I}}$, et chaque rôle à une partie de $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Ces trois éléments (concepts, rôles et individus) peuvent se combiner par l'intermédiaire de constructeurs pour former des descriptions de concepts et de rôles plus complexes qu'un simple nom. Les descriptions élémentaires sont appelées concepts et rôles atomiques (ce ne sont que des noms). Chaque logique de description est définie par sa propre combinaison de constructeurs. On peut citer :

- la conjonction de concepts notée $C \sqcap D$ qui permet de créer l'intersection des descriptions de concept C et D ;
- la quantification existentielle (resp. universelle), notée $\exists R.C$ (resp. $\forall R.C$) avec R un nom de rôle et C une description de concept, qui permet de définir l'ensemble des individus liés par R à au moins un individu lui-même appartenant à C (resp. liés par R uniquement à des individus de C);
- la négation, notée $\neg C$ permettant de définir la description de concept regroupant tous les individus qui ne sont pas dans C .

Par exemple, $Regle \sqcap \exists hasMaterial.Steel$ décrit l'ensemble des règles dont au moins un des matériaux le constituant est l'acier. Par ailleurs $Regle \sqcap \forall hasMaterial.Steel$ décrit l'ensemble des règles dont le matériau les constituant est l'acier.

Le tableau 1 regroupe la syntaxe et la sémantique des constructeurs de deux logiques de description : \mathcal{ALU} et \mathcal{EL}^+ . \mathcal{EL}^+ est une logique de description introduite dans ce travail et qui est égale à la logique \mathcal{EL}^{++} étudiée dans

[2] sans le constructeur des domaines concrets. Nous justifions plus bas l'intérêt de ce langage.

| Construct. | Syntaxe | Semantique |
|------------------------|---------------|--|
| concept atomique | P | $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ |
| rôle atomique | R | $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ |
| top | \top | $\Delta^{\mathcal{I}}$ |
| bottom | \perp | \emptyset |
| nominal | $\{a\}$ | $\{a^{\mathcal{I}}\}$ |
| conjonction | $C \sqcap D$ | $C^{\mathcal{I}} \cap D^{\mathcal{I}}$ |
| négation atomique | $\neg P$ | $\Delta^{\mathcal{I}} \setminus P^{\mathcal{I}}$ |
| quantif. universelle | $\forall R.C$ | $\{x \in \Delta^{\mathcal{I}} \mid \forall y : (x, y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$ |
| quantif. existentielle | $\exists R.C$ | $\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$ |
| disjonction | $C \sqcup D$ | $C^{\mathcal{I}} \cup D^{\mathcal{I}}$ |
| composition de rôles | $R \circ S$ | $\{(x, z) \in (\Delta^{\mathcal{I}})^2 \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in R^{\mathcal{I}} \wedge (y, z) \in S^{\mathcal{I}}\}$ |

TABLE 1 – Syntaxe et sémantique des constructeurs définissant la LD $\mathcal{EL}_{\setminus D}^{++}$, avec P un concept atomique, C et D des descriptions de concepts, R et S des rôles atomiques et a un nom d'individu.

Les descriptions de concepts peuvent être reliées entre elles par des axiomes, notamment de subsomption. Les axiomes de subsomption (appelés GCI pour global concept inclusion), notés $C \sqsubseteq D$, où C et D sont des descriptions de concept spécifiant que tous les individus de C sont aussi des individus de D . On peut diviser les instruments en deux catégories disjointes en fonction de leur mode de lecture (analogique ou numérique) grâce aux axiomes suivants :

$$Instrument \sqsubseteq analogicInstrument \sqcup numericInstrument$$

$$analogicInstrument \sqsubseteq \neg numericInstrument$$

Pour les rôles, on peut exprimer des axiomes d'inclusion de rôles du type $R \sqsubseteq S$, avec S un rôle atomique et R un rôle atomique ou une composition de rôles atomiques.

En plus des axiomes, on peut exprimer des assertions de concept (resp. de rôle), pour postuler que tel individu (resp. tel couple d'individus) est dans l'interprétation de telle description de concept (resp. de tel rôle). Les assertions $Ruler(rul1)$, $Steel(steel2\%)$ et $hasMaterial(rul1, steel2\%)$ décrivent que *rul1* est une règle, fait d'un acier particulier *steel2%*.

Un ensemble d'axiomes de concepts est appelé Terminological Box (TBox). Un ensemble d'axiomes de rôles est appelé Role Box (RBox). Une Constraint Box (CBox) est l'ensemble d'une TBox et d'une RBox. Un ensemble d'assertions est appelé Assertionnal Box (ABox). Dans la suite, on pourra utiliser le terme d'ontologie pour parler d'une

CBox, d'une CBox ou d'une ABox. La table 2 donne la liste des axiomes et assertions disponibles dans $\mathcal{EL}_{\mathcal{D}}^{++}$.

| Axiome et assertions | Syntaxe et Sémantique | $\mathcal{EL}_{\mathcal{D}}^{++}$ |
|---------------------------|--|-----------------------------------|
| General Concept Inclusion | $C \sqsubseteq D$ $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ | X |
| Inclusion de rôle | $R_1 \circ \dots \circ R_n \sqsubseteq R$ $R_1^{\mathcal{I}} \circ \dots \circ R_n^{\mathcal{I}} \subseteq r^{\mathcal{I}}$ | X |
| Assertion de concept | $C(a)$ $a^{\mathcal{I}} \in C^{\mathcal{I}}$ | X |
| Assertion de rôle | $R(a, b)$ $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ | X |

TABLE 2 – Syntaxe et sémantique des axiomes et assertions définissant la LD $\mathcal{EL}_{\mathcal{D}}^{++}$, C et D des $\mathcal{EL}_{\mathcal{D}}^{++}$ -descriptions de concepts, R, R_1, \dots, R_n des rôles atomiques et a et b des noms d'individus.

Exemple 1 (Exemple de CBox en métrologie). *La CBox ci-dessous décrit un univers métrologique composé d'instruments et de mesures. Une mesure est décrite par une unité et une dimension, un instrument par son matériau constituant, son type et son mode de lecture. Le reste des axiomes permet de créer une hiérarchie des concepts. On spécifie ainsi que le fer est un métal, et que le métal est un matériau (et donc par raisonnement que le fer est un matériau). Nous utiliserons cette ontologie dans les exemples qui suivront et elle permettra de décrire des ressources parlant de mesure et de l'instrument utilisé pour celle-ci.*

| | |
|--------------------|--|
| <i>Measurement</i> | $\sqsubseteq \exists \text{hasUnit. Unit} \sqcap$ $\exists \text{hasDimension. Dimension}$ |
| <i>Instrument</i> | $\sqsubseteq \exists \text{hasMaterial. Material} \sqcap$ $\exists \text{hasInstrumentType. InstrumentType} \sqcap$ $\exists \text{hasReadingMode. ReadingMode}$ |
| <i>Metal</i> | $\sqsubseteq \text{Material}$ |
| <i>Steel</i> | $\sqsubseteq \text{Metal}$ |
| <i>Iron</i> | $\sqsubseteq \text{Metal}$ |
| <i>Wood</i> | $\sqsubseteq \text{Material}$ |
| <i>Oak</i> | $\sqsubseteq \text{Wood}$ |
| <i>Analogic</i> | $\sqsubseteq \text{ReadingMode}$ |
| <i>Numeric</i> | $\sqsubseteq \text{ReadingMode}$ |
| <i>Length</i> | $\sqsubseteq \text{Dimension}$ |
| <i>Centimeter</i> | $\sqsubseteq \text{Unit}$ |
| <i>Ruler</i> | $\sqsubseteq \text{InstrumentType}$ |
| <i>Calliper</i> | $\sqsubseteq \text{InstrumentType}$ |

Pour faciliter la présentation des exemples, les termes de métrologie *Material*, *Dimension*, *InstrumentType* et *ReadingMode* seront abrégées en *Mat*, *Dim*, *IT* et *RM*. On utilise deux raisonnements de base avec les LD, la satisfaisabilité et la subsumption. Le premier vérifie qu'une interprétation \mathcal{I} satisfait l'ensemble des axiomes d'une CBox et des assertions d'une ABox. Le second consiste à vérifier que pour toute interprétation \mathcal{I} , on a : $C \sqsubseteq D \Leftrightarrow C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. On lit ainsi " C est subsumé par D ". Si C est subsumé

par D et D est subsumé par C , alors C et D sont équivalents, et on note $C \equiv D$. Si C est subsumé par D mais n'est pas équivalent à D , alors on note $C \sqsubset D$. On pourra aussi utiliser les notations inverses \sqsupseteq et \sqsupset . Dans cet article, on raisonnera toujours par rapport à une CBox CB non vide et une ABox AB (potentiellement vide).

Plus une LD possède de constructeurs, plus elle est expressive et se voit capable de modéliser finement le domaine qu'elle doit représenter. Cela s'accompagne néanmoins d'une complexité croissante lors de la résolution des raisonnements de base. Une difficulté de l'utilisation des LD réside donc dans leur choix, pour garantir un maximum d'expressivité tout en restant performant.

Dans ce travail, nous avons décidé d'utiliser la LD $\mathcal{EL}_{\mathcal{D}}^{++}$, qui est égale à \mathcal{EL}^{++} [2] privée des domaines concrets. En effet, il a été prouvé dans [2], que \mathcal{EL}^{++} conserve un temps polynomial pour la détermination de la subsumption et de la satisfaisabilité (sauf pour certains domaines concrets particuliers) en présence d'une CBox et d'une ABox, tout en ayant un pouvoir expressif conséquent, contrairement à d'autres LD basées sur l'utilisation de la quantification existentielle. Par ailleurs, \mathcal{EL}^{++} est également munie d'une forme normale définie dans [2] ce qui en facilite l'utilisation.

\mathcal{EL}^{++} est connue pour être suffisamment expressive pour un grand nombre d'applications, ce que nous avons vérifié dans la construction de l'ontologie de la métrologie pour le projet STAM. N'ayant pas eu besoin pour le moment des domaines concrets, nous limitons notre étude à $\mathcal{EL}_{\mathcal{D}}^{++}$. Enfin, un dernier atout de cette LD est qu'elle est proche du profil \mathcal{EL} de OWL2, le langage standard pour la modélisation d'ontologie sur le web, ce qui rend possible l'usage des outils associés existants (éditeurs et raisonneurs).

3 Recherche des ressources proches

Nous détaillons dans cette section notre approche pour trouver les ressources les plus proches d'une requête donnée. Nous commençons en section 3.1 par expliquer la structure en composantes des descriptions des ressources et de la requête. En section 3.2, nous montrons comment classer les ressources par rapport à chaque composante avec un raisonnement en LD. En section 3.3, nous proposons un classement des ressources qui intègre toutes les composantes par une approche multicritère.

3.1 Recherche avec composante

Nous postulons que les requêtes et ressources sont décrites selon plusieurs dimensions que l'on appelle composantes. Cette structure de description offre l'avantage de diviser la comparaison globale d'une requête avec une ressource en plusieurs comparaisons (une par composante) plus simples à effectuer, sur le principe "diviser pour régner".

Exemple 2. *Un document pdf nommé DocCaliper1 décrivant une mesure de diamètre avec un pied à coulisse numérique en acier peut être décrit par une composante spécifiant le type d'instrument (introduite par le rôle *hasInstrument*) et une composante spécifiant la mesure*

(introduite par le rôle *hasMeasure*) :

$$\begin{aligned} \text{DocCaliper1} \equiv & \exists \text{hasInstrument}. \\ & (\exists \text{hasIT.Calliper} \sqcap \\ & \exists \text{hasRM.Numeric} \sqcap \\ & \exists \text{hasMat.Steel}) \\ & \sqcap \\ & \exists \text{hasMeasure}. \quad (\exists \text{hasUnit.cm} \sqcap \\ & \exists \text{hasDim.length}) \end{aligned}$$

La recherche consiste ainsi dans un premier temps à classer les ressources les plus proches de la requête pour chaque composante, puis à agréger ces classements par composante pour en obtenir un unique global. Cette structure en composantes est obtenue, pour les ressources, grâce aux experts qui les décrivent, et pour les requêtes grâce à un processus qui n'est pas développé dans cet article permettant de générer automatiquement les descriptions par composante des requêtes utilisateur à partir des facettes et des mots-clés choisis par ces derniers. Par la suite, nous désignerons la requête de l'utilisateur par le terme "demande", et les ressources disponibles par le terme "offres".

Nous formalisons maintenant notre approche à base de composantes.

Definition 1 (Composante). *Etant donnée une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox CB et E une description de concept de CB, la composante \mathcal{C}_E est l'ensemble des $\mathcal{EL}_{\mathcal{D}}^{++}$ -descriptions de concepts qui sont subsumées par E dans CB (avant et après inférence). E est appelée le top concept de \mathcal{C}_E .*

Les composantes sont a priori toutes indépendantes (deux à deux), i.e. les rôles de composantes sont orthogonaux, i.e. pas de subsomption entre rôles de composantes deux à deux. Cette hypothèse provient de l'application STAM et de la nature des requêtes envisagées.

On suppose que, pour chaque composante \mathcal{C}_E d'une CBox CB, il existe un rôle $R_{\mathcal{C}_E}$ dit "rôle de composante" associé à \mathcal{C}_E . On dira aussi que \mathcal{C}_E est associée à $R_{\mathcal{C}_E}$. On suppose que le top concept E de \mathcal{C}_E est aussi la portée du rôle $R_{\mathcal{C}_E}$.

Exemple 3. *Avec la CBox de l'exemple 1, on définit deux composantes : Instrument et son top concept Instrument, et Measure avec son top concept Measure. On suppose l'existence dans la CBox deux rôles de composante associés *hasInstrument* et *hasMeasure* (cf exemple 2).*

Par commodité de langage et quand le contexte sera clair, on pourra employer le terme composante soit dans le sens exact de sa définition, soit pour évoquer son top concept, soit pour évoquer son rôle de composante associé.

Definition 2 (Offre (resp. demande)). *Etant donnée une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox CB et $\mathcal{C}_{E_1}, \dots, \mathcal{C}_{E_n}$ les n composantes de CB (données arbitrairement), avec R_1, \dots, R_n les rôles de composantes associés, une offre O (resp. une demande D) est une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description de concept qui s'écrit $O \equiv \exists R_1.C_1 \sqcap \dots \sqcap \exists R_n.C_n$ où chaque C_i une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description de concept appartenant à \mathcal{C}_{E_i} , $\forall i \in \{1, \dots, n\}$.*

Toute offre et demande admet dans sa description un terme pour chaque composante. Si la composante R n'est pas utile pour la description, alors elle est fixée à $\exists R.\top$.

Exemple 4. *En reprenant la CBox de l'exemple 1 avec les composantes de l'exemple 3, on peut imaginer les demandes D1 et D2 qui recherchent des ressources sur des pieds à coulisse numériques en acier, et des règles en bois graduées en cm :*

$$\begin{aligned} D1 \equiv & \exists \text{hasInstrument}.(\exists \text{hasIT.Calliper} \sqcap \\ & \exists \text{hasRM.Numeric} \sqcap \\ & \exists \text{hasMat.}(Steel)) \sqcap \\ & \exists \text{hasMeasure}.\top \end{aligned}$$

$$\begin{aligned} D2 \equiv & \exists \text{hasInstrument}.(Ruler \sqcap \exists \text{hasMat.Wood}) \\ & \sqcap \exists \text{hasMeasure}.(\exists \text{hasUnit.Centimeter}) \end{aligned}$$

Definition 3 (Projection sur une composante). *Soient une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox CB, $\mathcal{C}_{E_1}, \dots, \mathcal{C}_{E_n}$ les n composantes de CB (données arbitrairement), avec R_1, \dots, R_n les rôles de composantes associés, et une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description de concept $O \equiv \exists R_1.C_1 \sqcap \dots \sqcap \exists R_n.C_n$. La projection de O sur \mathcal{C}_{E_i} est la description de concept C_i , $\forall i \in \{1, \dots, n\}$. On la note O^{R_i} .*

Au sein d'une demande ou d'une offre, selon la projection correspondante, les composantes sont considérées inexistantes ou existantes, selon la définition suivante.

Definition 4 (Composante existante et inexistante). *Etant donnée une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox CB, \mathcal{C}_E une composante donnée, $R_{\mathcal{C}_E}$ le rôle de composante associé, et O une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description de concept. On dit que \mathcal{C}_E est existante dans O si $O^{R_{\mathcal{C}_E}} \neq \top$, et inexistante sinon,*

Si $\exists R_1.C_1$ n'apparaît pas dans la description de concept, alors $\exists R_1.\top$ y apparaît pour cette définition. Même si l'ajout de cette description change la sémantique d'une offre ou d'une demande, c'est un pré-traitement au calcul de proximité qui ne change pas le résultat de celui-ci.

3.2 Classement sémantique par composante

Nous proposons maintenant de classer les offres selon leur proximité sémantique par rapport à la demande, pour chaque composante. On doit ici formaliser la notion de plus grande proximité entre une offre et une demande, pour une composante donnée. On s'inspire de la notion de meilleure couverture [9] pour définir la plus grande proximité entre une offre et une demande par la maximisation de l'information commune entre elles. On obtient cela en minimisant respectivement les informations de la demande absentes de l'offre d'un côté, et les informations de l'offre absentes de la demande de l'autre. Ce sont les notions de Rest et Miss, basées sur le calcul de l'information commune entre l'offre et la demande (raisonnement de plus petit subsumant commun ou least common subsumer [1]), et sur le calcul de l'information de l'une manquante dans l'autre (différence sémantique [10]).

Exemple 5. *Soient D une demande et O une offre :*

$$\begin{aligned}
 D &\equiv \exists \text{hasInstrument}(\exists \text{hasIT.Calliper} \sqcap \\
 &\quad \exists \text{hasMat.Steel}) \sqcap \\
 &\quad \exists \text{hasMeasure}(\exists \text{hasDim.length} \sqcap \\
 &\quad \exists \text{hasUnit.centimeter}) \\
 O &\equiv \exists \text{hasInstrument}(\exists \text{hasIT.Calliper} \sqcap \\
 &\quad \exists \text{hasMat.Metal} \sqcap \exists \text{hasRM.Numeric}) \sqcap \\
 &\quad \exists \text{hasMeasure}(\exists \text{hasDim.length})
 \end{aligned}$$

Intuitivement, pour la composante *hasInstrument*, on voudrait que le Rest qui est la partie de la demande non couverte par l'offre soit $\exists \text{hasMat.Steel}$, et que le Miss qui est la partie de l'offre qui n'est pas demandée dans la demande soit $\exists \text{hasRM.Numeric}$. Pour la composante *hasMeasure*, on voudrait que le Rest soit $\exists \text{hasUnit.centimeter}$, et que le Miss soit \top (c'est-à-dire qu'il n'y ait pas de Miss).

Pour déterminer le Rest (resp. le Miss), [9] suggère d'oter à la demande (resp. à l'offre) les informations communes aux deux. On rappelle donc les définitions de least common subsumer (LCS) et de différence sémantique permettant de réaliser ces opérations dans les logiques de description.

Definition 5 (Least Common Subsumer, lcs [1]). *Soient C et D deux descriptions de concepts appartenant à la logique \mathcal{L} . La description de concept E est un plus petit subsumant commun de C et D si et seulement si : (a) $C \sqsubseteq E$ et $D \sqsubseteq E$, et (b) E est le concept le plus spécifique (le plus petit par rapport à la subsumption) à respecter (a).*

Quand le LCS de C et D existe, il est souvent unique et noté $LCS(C, D)$, ou $LCS_{CB}(C, D)$ s'il est calculé par rapport à une CBox CB. Dans ce travail, encore en cours, nous faisons l'hypothèse que le LCS existe et est unique dans $\mathcal{EL}_{\mathcal{D}}^{++}$. Cela reste à démontrer. Il existe cependant dans \mathcal{EL}^+ et si aucun calcul de complexité n'a été fait, des tests pratiques ont réalisés. Il est néanmoins nécessaire de poser une limite de profondeur d'exploration des rôles, en cas de cycle. Le calcul se fait par l'intermédiaire d'un graphe de complétion réalisé à partir d'une TBox normalisée. Une étape supplémentaire de dénormalisation et de simplification permet ensuite d'extraire le LCS de l'ensemble des éléments communs de deux descriptions de concept. [3]

Exemple 6. *En reprenant la CBox de l'exemple 1, on définit les concepts A , B et C ainsi :*

$A \equiv \text{Steel} \sqcap \text{Analogic}$; $B \equiv \text{Iron} \sqcap \text{Numeric}$; $C \equiv \text{Oak}$
 Ainsi, $LCS_{CB}(A, B) \equiv \text{Metal} \sqcap \text{RM}$ puisque l'on a dans la Cbox $\text{Steel} \sqsubseteq \text{Metal}$ et $\text{Iron} \sqsubseteq \text{Metal}$ et puisque $\text{Analogic} \sqsubseteq \text{RM}$ et $\text{Numeric} \sqsubseteq \text{RM}$.

De la même manière $LCS_{CB}(A, C) \equiv \text{Mat}$ et $LCS_{CB}(B, C) \equiv \text{Material}$ car dans la CBox, Mat est le concept le plus précis subsumant à la fois Oak et Steel (resp. Oak et Iron) (et Analogic et Numeric sont subsumés par \top .)

Definition 6 (Différences sémantiques). *Soient deux descriptions de concepts C et D appartenant à la logique \mathcal{L} , avec $C \sqsubseteq D$. La différence sémantique de Teege [10] $C \ominus D$ est définie par : $\max_{\sqsubseteq} \{E \in \mathcal{L} \mid E \sqcap D \equiv C\}$.*

La différence de Suchanek [4] est définie ainsi : Soit un concept A et une description de concept C ordonnée selon la forme normale proposée dans la publication $\text{norm}(C) = C_1 \sqcap \dots \sqcap C_n$, la soustraction $C - A := \text{norm}(C_1 \sqcap \dots \sqcap C_{j-1} \sqcap C_{j+1} \sqcap \dots \sqcap C_n)$, avec $C_j \sqsubseteq A$ et j le plus petit possible. S'il n'existe pas un tel j , alors $C - A = C$. Si A est une conjonction de concept, $C - A = C - A_1 - \dots - A_m$.

La différence de Heinz [6] reprend le principe de la différence de Suchanek en définissant un ensemble de propriétés, lui permettant de retirer des descriptions de concept à l'intérieur d'un quantificateur existentiel, ce que les deux différences précédentes ne sont pas capables de faire.

La différence sémantique peut ne pas être unique [9]. Comme le LCS, nous faisons ici l'hypothèse que la différence sémantique est unique dans $\mathcal{EL}_{\mathcal{D}}^{++}$. Cela reste à démontrer. On définit maintenant les Rest et Miss.

Definition 7 (Rest et Miss). *Soient CB une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox, et O et D deux $\mathcal{EL}_{\mathcal{D}}^{++}$ -descriptions de concepts. Le Rest de D par O et le Miss de D par O sont notés $\text{Rest}_D(O)$ et $\text{Miss}_D(O)$ et sont définis ainsi :*
 $\text{Rest}_D(O) \equiv D \ominus LCS_{CB}(D, O)$
 $\text{Miss}_D(O) \equiv O \ominus LCS_{CB}(D, O)$

Exemple 7. *Soit CB la CBox de l'exemple 1. Soient D et O les descriptions : $D \equiv \text{Steel} \sqcap \text{Analogic}$ et $O \equiv \text{Metal} \sqcap \text{Numeric}$. On a : $LCS_{CB}(D, O) \equiv \text{Metal} \sqcap \text{RM}$
 $\text{Rest}_D(O) \equiv (\text{Steel} \sqcap \text{Analogic}) \ominus (\text{Metal} \sqcap \text{RM}) \equiv \text{Steel} \sqcap \text{Analogic} \equiv D$
 $\text{Miss}_D(O) \equiv (\text{Metal} \sqcap \text{Numeric}) \ominus (\text{Metal} \sqcap \text{RM}) \equiv \text{Numeric}$
 Ainsi O ne couvre aucune information de D et ajoute des informations superflues (le mode de lecture Numeric).*

Nous pouvons maintenant proposer une méthode de classement des offres par rapport à une demande. Tout d'abord, on ne classe que les offres qui ont au moins une composante renseignée commune avec la demande, offres que l'on appellera recommandations. Afin de profiter de la structure par composante des recommandations et de la demande, comme évoqué précédemment, on classe les recommandations par rapport à chaque composante de la CBox. Pour chaque composante, les recommandations les plus proches de la demande sont celles qui optimisent le Rest, puis en cas d'égalité celles qui optimisent le Miss. Optimiser le Rest en premier permet d'assurer que le plus possible d'informations données dans la demande soient présentes dans les meilleures recommandations. Optimiser le Miss sert à départager les recommandations qui seraient semblables du point de vue du Rest. L'optimisation des Rest et Miss va consister dans un premier temps à les maximiser par rapport à la subsumption (puisque plus un concept est grand par rapport à la subsumption, plus il est général et moins il contient d'information). Dans un second temps, là-encore pour départager les ex-aequo, on optimisera Rest et Miss en minimisant leur longueur syntaxique.

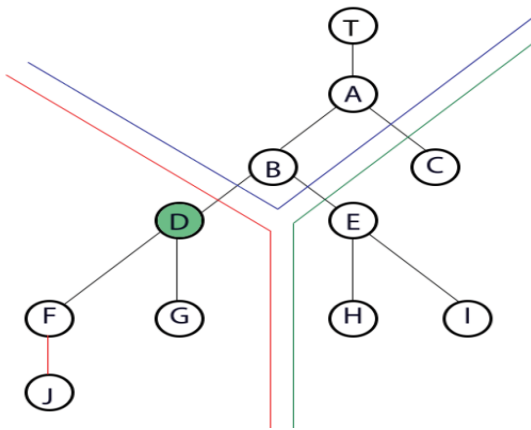


FIGURE 1 – Exemple de positionnement des projections de recommandations (A, B, C, E, F, G, H, I et J) et d'une demande (D) sur une composante. Les arêtes sont des liens de subsomption, et plus on monte, plus on est grand par rapport à la subsomption. Le concept en haut est T .

Definition 8 (Longueur syntaxique d'une description). Soit D une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description de concept. La longueur syntaxique de D , notée $|D|$ est le nombre de concepts atomiques contenus dans D .

La figure 1 illustre les 4 cas possibles qui peuvent survenir entre une demande D et une recommandation, pour une composante. Dans la zone en rouge (en bas à gauche) se trouve la projection de D sur la composante (rond en vert contenant D). F, G et J sont des projections possibles sur cette composante : ces projections sont subsumées par celle de D . On dit alors que la recommandation est plus précise que la demande. Dans la zone en bleu (en haut), on a les cas A, B et T de projections de recommandation qui subsument celle de la demande. On dit alors que la recommandation est moins précise que la demande. Enfin dans la zone en vert (en bas à droite), C, E, H et I sont des cas où la recommandation est dite éloignée de la demande.

L'algorithme 2 permet de comparer les recommandations 2 à 2 par rapport à une demande, selon leur zone d'appartenance dans l'arbre (cf. figure 1). Il est facile de montrer que plus on monte dans cet arbre, plus le Rest est petit par rapport à la subsomption (i.e. moins il est bon), et plus on descend, plus le Miss est petit par rapport à la subsomption. Ainsi les recommandations plus précises maximisent le Rest (puisque ce dernier est alors T) et sont donc meilleures que les recommandations moins précises, elles-mêmes meilleures que les recommandations éloignées.

A l'intérieur de chaque zone, on a les situations suivantes :
- si deux recommandations sont plus précises alors on les départage en comparant leur miss : le miss le plus général (i.e. grand par rapport à la subsomption) est meilleur, et si les miss sont équivalents ou incomparables par rapport à la subsomption, le meilleur miss est le plus petit en taille . En cas de taille identique, les deux offres sont considérées équivalentes.

- si deux recommandations sont moins précises alors on

les départage en comparant leur rest : le rest le plus général par rapport à la subsomption est meilleur, et si les rest sont équivalents ou incomparables par rapport à la subsomption, le meilleur rest est le plus petit en taille. En cas de taille identique, les deux offres sont considérées équivalentes.

- si deux recommandations sont éloignées, alors on cherche en premier celle qui maximise le rest par rapport à la subsomption, puis qui minimise le rest en taille, et en second si le rest n'a pu les départager, celle qui maximise le miss par rapport à la subsomption puis qui minimise le miss en taille. Si ces critères ne suffisent pas à départager les deux offres, alors elles sont considérées équivalentes.

3.3 Approche multicritère

Notre objectif est maintenant de classer les recommandations par rapport à toutes les composantes, sachant qu'on a vu à la section précédente comment les classer deux à deux par rapport à chaque composante. Comme la conjonction de deux concepts EL^+ est également un concept EL^+ , on pourrait définir la similitude entre concepts sur EL^+ et de l'appliquer directement à la conjonction des concepts EL^+ représentant l'ensemble des composantes. Cependant on risquerait ainsi d'accorder plus d'importance à telle ou telle composante en fonction de la quantité d'informations utilisée pour les décrire. C'est pour éviter de privilégier une composante par rapport à une autre que l'on compare ces dernières deux à deux. Autrement dit, la répartition des scores sur des composantes décidées par l'expert du domaine permet ainsi de donner une importance équivalente à une description de concept simple mais néanmoins d'importance égale.

On voit facilement que l'on a à faire ici à un problème d'ordonnement multicritère, où chaque composante est un critère. Nous proposons d'utiliser une relation de concordance relative [7] pour le résoudre. C'est un des procédés les plus simples en ordonnancement multicritère et nous permet d'intégrer facilement des critères entre composante qui ne sont pas directement numériques (puisque la comparaison de deux composantes est basée sur le LCS et la différence sémantique). D'autres raisonnements d'ordonnement multicritère pourraient être utilisés à condition de traduire de façon numérique la proximité sémantique, ce qui semble plus complexe. Cela pourrait permettre d'exploiter l'ensemble des méthodes d'ordonnement multicritère.

Nous rappelons maintenant le principe de la concordance relative.

Chaque recommandation x est comparée avec toutes les autres recommandations y sur chaque composante i , $1 \leq i \leq n$, par la fonction $\phi_i(x, y)$ définie ci-dessous, qui nous donne le score relatif de x par rapport y pour i .

$$\phi_i(x, y) = \begin{cases} si & x_i > y_i, & \phi_i(x, y) = 1 \\ si & x_i = y_i, & \phi_i(x, y) = 0 \\ si & x_i < y_i, & \phi_i(x, y) = -1 \end{cases}$$

On obtient alors le score relatif $c(x, y)$ de x par rapport y pour toutes les n composantes en faisant la somme des $\phi_i(x, y)$:

$$c(x, y) = \sum_{i=0}^n \phi_i(x, y)$$

On a bien entendu $c(x, y) = -c(y, x)$. Et on dira que x est meilleure que y ssi $c(x, y) \geq c(y, x)$ (ou $c(x, y) \geq 0$). Pour obtenir le score global de x , on fait ensuite la somme pour toutes les autres recommandations y de $c(x, y)$:

$$score(x) = \sum_{y, y \neq x} c(x, y)$$

Il suffit ensuite d'ordonner de façon décroissante les scores globaux pour classer les meilleures recommandations. Notons qu'il est possible de donner plus d'importance à des composantes i en particulier en ajoutant un coefficient v_i lors du calcul de ϕ_i .

Exemple 8. On suppose que l'on a 3 recommandations x, y et z , et 3 composantes $i, 1 \leq i \leq 3$. On suppose de plus que chaque recommandation possède une valeur pour chaque composante, ce qui permet de comparer les recommandations deux à deux pour chaque composante. Les valeurs de x sont (10, 5, 8), de y sont (11, 4, 8) et de z sont (9, 3, 7). Ainsi, les scores relatifs des recommandations pour chaque composante sont :

$$\begin{aligned} c(x, y) &= \phi_1(10, 11) + \phi_2(5, 4) + \phi_3(8, 8) = -1 + 1 + 0 = 0 \\ c(y, z) &= \phi_1(11, 9) + \phi_2(4, 3) + \phi_3(8, 7) = 1 + 1 + 1 = 3 \\ c(x, z) &= \phi_1(10, 9) + \phi_2(5, 3) + \phi_3(8, 7) = 1 + 1 + 1 = 3 \end{aligned}$$

Ainsi, x et y sont équivalentes, et z est moins bonne, car $score(x) = score(y) = 3$ et $score(z) = -6$.

Supposons maintenant que l'on veuille privilégier une composante en lui attribuant un coefficient de 3 (en laissant un coefficient de 1 aux deux autres). On a donc :

$$\begin{aligned} c(x, y) &= \phi_1(10, 11) + \phi_2(5, 4) + \phi_3(8, 8) = -3 + 1 + 0 = -2 \\ c(y, z) &= \phi_1(11, 9) + \phi_2(4, 3) + \phi_3(8, 7) = 3 + 1 + 1 = 5 \\ c(x, z) &= \phi_1(10, 9) + \phi_2(5, 3) + \phi_3(8, 7) = 3 + 1 + 1 = 5 \end{aligned}$$

Ainsi, y est meilleure que x et elles sont toutes deux meilleures que z , car $score(y) = 7$, $score(x) = 3$ et $score(z) = -10$.

Il est facile d'appliquer le principe de concordance relative à notre cas d'étude, puisque l'algorithme 2 implémente une fonction ϕ_i , appelée ici $\phi_{R, CB}(D)(O_1, O_2)$: les composantes R de la CBox CB sont les composantes, D est la demande et O_1 et O_2 sont les recommandations à comparer par rapport à R . Contrairement à l'exemple précédent, $\phi_{R, CB}(D)(O_1, O_2)$ compare O_1 et O_2 sur des critères sémantiques et non numériques.

Exemple 9 (Demandes et offres). En prenant la CBox CB de l'exemple 1, soit D la demande utilisateur, O_1, O_2, O_3, O_4 , les offres de notre base de connaissance. Les rôles de composantes sont abrégés en R_1 pour $hasInstrument$ et R_2 pour $hasMeasure$.

$$D \equiv \exists R_1. (\exists hasMat.Metal \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \sqcap \exists R_2. (\exists hasUnit.Centimeter \sqcap \exists hasDim.T)$$

$$O_1 \equiv \exists R_1. (\exists hasMat.Steel \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \sqcap \exists R_2. (\exists hasUnit.Centimeter \sqcap \exists hasDim.T)$$

Algorithme 1 Tri des recommandations

Entrée(s) : Une $\mathcal{EL}_{\mathcal{D}}^{++}$ -CBox CB avec C_{E_1}, \dots, C_{E_n} les n composantes de CB (données arbitrairement) et R_1, \dots, R_n les rôles de composantes associés, une $\mathcal{EL}_{\mathcal{D}}^{++}$ -description D (la demande), et m $\mathcal{EL}_{\mathcal{D}}^{++}$ -descriptions O_1, \dots, O_m (les offres).

Sortie(s) : L'ensemble $e_O = \{(O_j, score_j) \mid j \in \{1, \dots, m\}\}$ des couples composés de O_j et du score associé $score_j$ par rapport à D , classés du plus grand score au plus petit.

- 1: $e_O := \emptyset$
 - 2: Initialisation de $score_1$ à $score_m$ à 0
 - 3: **pour chaque** $(O_i, O_j) \in \{O_1, \dots, O_m\}^2$ avec $j > i$ **faire**
 - 4: **pour chaque** composante $R_k \in \{R_1, \dots, R_n\}$ **faire**
 - 5: $score_i := score_i + \phi_{R_k, CB}(D)(O_i, O_j)$ //cf. algo. 2
 - 6: $score_j := score_j - \phi_{R_k, CB}(D)(O_i, O_j)$
 - 7: **fin pour**
 - 8: **fin pour**
 - 9: **pour chaque** O_j **faire**
 - 10: $e_O := e_O \cup (O_j, score_j)$
 - 11: **fin pour**
 - 12: Tri des couples de e_O par score décroissant.
 - 13: renvoyer e_O
-

$$O_2 \equiv \exists R_1. (\exists hasMat.oak \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \sqcap \exists R_2. (\exists hasUnit.Centimeter \sqcap \exists hasDim.T)$$

$$O_3 \equiv \exists R_1. (\exists hasMat.Metal \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \sqcap \exists R_2. (\exists hasUnit.T \sqcap \exists hasDim.T)$$

$$O_4 \equiv \exists R_1. (\exists hasMat.Wood \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.T) \sqcap \exists R_2. (\exists hasUnit.Centimeter \sqcap \exists hasDim.T)$$

Pour chaque couple d'offres, on compare leurs composantes grâce à l'algorithme 2. Prenons quelques exemples, avec (O_1, O_3) , (O_1, O_2) et (O_2, O_4) sur la composante R_1 avec $score_i$ le score calculé par l'algorithme :

$$- O_1^{R_1} \sqsubset D^{R_1} \text{ et } O_3^{R_1} \equiv D^{R_1}, \text{ alors } score_1 := score_1 - 1 \text{ et } score_3 := score_3 + 1$$

$$- O_1^{R_1} \sqsubset D^{R_1} \text{ et } O_2^{R_1} \not\sqsubset D^{R_1} \text{ et } O_2^{R_1} \not\equiv D^{R_1}, \text{ alors } score_1 := score_1 + 1 \text{ et } score_2 := score_2 - 1$$

$$- O_2^{R_1} \text{ et } O_4^{R_1} \text{ ne subsument et ne sont pas subsumées par } D^{R_1}, \text{ on compare donc leur Rest :}$$

$$Rest_{D^{R_1}}(O_2^{R_1}) \equiv (\exists hasMat.Metal \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \ominus (\exists hasMat.Oak \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \equiv \exists hasMat.Metal$$

$$Rest_{D^{R_1}}(O_4^{R_1}) \equiv (\exists hasMat.Metal \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.Analogic) \ominus (\exists hasMat.Wood \sqcap \exists hasIT.Ruler \sqcap \exists hasRM.T) \equiv \exists hasMat.Metal \sqcap \exists hasRM.Analogic$$

$$Rest_{D^{R_1}}(O_4^{R_1}) \sqsubset Rest_{D^{R_1}}(O_2^{R_1}) \text{ alors } score_4 := score_4 - 1 \text{ et } score_2 := score_2 + 1$$

On fait ainsi pour les deux composantes, chaque couple et on obtient les scores finaux :

$$score_1 = 2, score_2 = 0, score_3 = 0, score_4 = -2$$

A l'issue de l'algorithme 1, l'ensemble des offres avec leurs scores triées de façon décroissante est :

$$(O_1, 2), (O_2, 0), (O_3, 0), (O_4, -2).$$

O_1 est donc la recommandation la plus proche de D .

Algorithme 2 Fonction $\phi_{R,CB}(D)(O_1, O_2)$ d'ordonnement sémantique de deux recommandations O_1 et O_2 pour une demande D par rapport à une composante R de l'ontologie CB.

Entrée(s) : Une $\mathcal{EL}_{\setminus D}^{++}$ -CBox CB, un rôle de composante R de CB, la description D d'une demande, et les descriptions O_1 et O_2 de deux recommandations.

Sortie(s) : 1 si O_1 est meilleure que O_2 pour D dans CB p/r à R , -1 si O_2 est meilleure, et 0 si O_1 et O_2 sont équivalentes.

```

1: si  $O_1^R \sqsubset D^R$  alors
2:   si  $O_2^R \sqsubset D^R$  alors
3:     si  $Miss_{DR}(O_1^R) \sqsubset Miss_{DR}(O_2^R)$  alors
4:       Renvoyer -1
5:     sinon si  $Miss_{DR}(O_1^R) \sqsupset Miss_{DR}(O_2^R)$  alors
6:       Renvoyer 1
7:     sinon
8:       si  $|Miss_{DR}(O_1^R)| > |Miss_{DR}(O_2^R)|$  alors
9:         Renvoyer -1
10:      sinon si  $|Miss_{DR}(O_1^R)| < |Miss_{DR}(O_2^R)|$  alors
11:        Renvoyer 1
12:      sinon
13:        Renvoyer 0
14:      fin si
15:    fin si
16:  sinon si  $O_2^R \equiv D^R$  alors
17:    Renvoyer -1
18:  sinon
19:    Renvoyer 1
20:  fin si
21: sinon si  $O_1^R \sqsupset D^R$  alors
22:   si  $O_2^R \sqsupset D^R$  alors
23:     si  $Rest_{DR}(O_1^R) \sqsubset Rest_{DR}(O_2^R)$  alors
24:       Renvoyer -1
25:     sinon si  $Rest_{DR}(O_1^R) \sqsupset Rest_{DR}(O_2^R)$  alors
26:       Renvoyer 1
27:     sinon
28:       si  $|Rest_{DR}(O_1^R)| > |Rest_{DR}(O_2^R)|$  alors
29:         Renvoyer -1
30:       sinon si  $|Rest_{DR}(O_1^R)| < |Rest_{DR}(O_2^R)|$  alors
31:        Renvoyer 1
32:       sinon
33:        Renvoyer 0
34:       fin si
35:     fin si
36:   sinon si  $O_2^R \sqsubseteq D^R$  alors
37:     Renvoyer -1
38:   sinon
39:     Renvoyer 1
40:   fin si
41: sinon si  $O_1^R \equiv D^R$  alors
42:   si  $O_2^R \equiv D^R$  alors
43:     Renvoyer 0
44:   sinon
45:     Renvoyer 1
46:   fin si
47: sinon
48:   si  $(O_2^R \sqsubseteq D^R)$  ou  $(O_2^R \sqsupset D^R)$  alors
49:     Renvoyer -1
50:   sinon
51:     si  $Rest_{DR}(O_1^R) \sqsubset Rest_{DR}(O_2^R)$  alors
52:       Renvoyer -1
53:     sinon si  $Rest_{DR}(O_1^R) \sqsupset Rest_{DR}(O_2^R)$  alors
54:       Renvoyer 1
55:     sinon
56:       si  $|Rest_{DR}(O_1^R)| > |Rest_{DR}(O_2^R)|$  alors
57:         Renvoyer -1
58:       sinon si  $|Rest_{DR}(O_1^R)| < |Rest_{DR}(O_2^R)|$  alors
59:         Renvoyer 1
60:       sinon
61:         si  $Miss_{DR}(O_1^R) \sqsubset Miss_{DR}(O_2^R)$  alors
62:           Renvoyer -1
63:         sinon si  $Miss_{DR}(O_1^R) \sqsupset Miss_{DR}(O_2^R)$  alors
64:           Renvoyer 1
65:         sinon
66:           si  $|Miss_{DR}(O_1^R)| > |Miss_{DR}(O_2^R)|$  alors
67:             Renvoyer -1
68:           sinon si  $|Miss_{DR}(O_1^R)| < |Miss_{DR}(O_2^R)|$ 
69:             alors
70:               Renvoyer 1
71:             sinon
72:               Renvoyer 0
73:             fin si
74:           fin si
75:         fin si
76:       fin si
77:     fin si

```

4 Travaux antérieurs

Selon [5], les algorithmes de matchmaking sémantiques dépendent de deux caractéristiques : la présence ou non du calcul de la distance entre l'offre et la demande, et l'arité de la réponse. L'arité vaut 1-1 quand on classe plusieurs offres séparées, et elle est 1-N quand on classe plusieurs ensembles d'offres (pour mieux répondre à la demande). Dans [8] sont proposées des algorithmes de matchmaking basés sur les raisonnements d'abduction et de contraction de concept. L'abduction est ici utilisée pour ajouter à une offre les descriptions de concept qu'il lui manque pour satisfaire une demande, tout en respectant les axiomes de la TBox correspondante. A contrario, la contraction de concept permet de retirer les descriptions de concept dans

la demande dont la présence bloque la recommandation d'une offre. Les deux méthodes permettent de calculer une distance équivalente au nombre de modifications nécessaires pour passer de la demande à l'offre et vis-versa, afin de classer les différentes offres de la plus proche à la plus éloignée. Elles sont toutes deux d'arités 1-1, c'est-à-dire qu'elle mette en relation une demande avec une seule offre. Le principe de meilleures couvertures[9] est différent, puisqu'il va mettre en relation une demande avec un ensemble d'offre dont la conjonction, appelée couverture, permet de répondre au maximum à la demande, il est donc d'arité 1-n. Le calcul de distance entre la demande et les ensembles d'offres utilise les concept de Rest et de Miss, qui correspondent respectivement aux caractéristiques que la couverture ne couvre pas dans la demande et les caractéristiques

rajoutées par les réponses qui ne se trouvaient pas dans la demande. Notre méthode de recommandation sémantique propose de reprendre le principe de Rest et de Miss pour le calcul des distances mais de l'appliquer à un système d'arité 1-1. On remplace un matchmaking d'arité 1-N par un matchmaking d'arité 1-1 mais sur plusieurs composantes. On passe alors d'une recherche combinatoire à un problème d'ordonnement multicritère et nous espérons dans le futur prouver que cela fait baisser la complexité des algorithmes. Par ailleurs, les approches multicritères sont facilement personnalisables (avec par exemple l'ajout de coefficients pour pondérer les composantes) et donc adaptables à de nombreux contextes.

5 Conclusion

Dans le contexte du projet STAM dans le domaine de la métrologie, nous nous sommes intéressés au problème du classement de recommandations pour une requête utilisateur donnée. Nous avons formalisé la notion de proximité d'une recommandation par rapport à la requête : à partir des composantes, que nous avons définies comme les différentes parties de la requête et des recommandations, nous avons proposé un raisonnement dans la LD \mathcal{EL}^+ permettant de définir les recommandations retenues et de les comparer entre elles, pour obtenir un classement des recommandations pour chaque composante. Ce raisonnement s'appuie sur les raisonnements de différence et de LCS. Puis nous avons établi un ordonnancement global des recommandations selon une approche multicritère permettant d'intégrer les classements de toutes les composantes. Ce travail est encore en cours. Comme évoqué précédemment, il reste à démontrer l'existence d'un LCS et d'une différence sémantique unique pour deux \mathcal{EL}^+ -descriptions par rapport à une \mathcal{EL}^+ -CBox. De plus, l'étude de complexité théorique et calculatoire n'a pas été entamée. Enfin, nous envisageons de faire des évaluations en termes de précision et de rappel au sein du projet STAM.

Remerciements

Ce travail a été financé par le FEDER https://ec.europa.eu/regional_policy/fr/funding/erdf.

Références

- [1] F. Baader. Computing the least common subsumer in the description logic el w.r.t. terminological cycles with descriptive semantics. In *ICCS*, 2003.
- [2] F. Baader, S. Brandt, and C. Lutz. Pushing the el envelope. In *International Joint Conferences on Artificial Intelligence (IJCAI-05)*, 2005.
- [3] A. Ecke. Completion-based role-depth bounded least common subsumer for extensions of el, 2012.
- [4] F. Suchanek, C. Menar, M. Bienvenu, and C. Chapelier. Technical report : a language for combinatorial creativity. Technical report, Telecom ParisTech, sep 2016.
- [5] M.-S. Hacid, F. Lécué, A. Léger, C. Rey, and F. Toumani. Les web services sémantiques, automate et

intégration. *Technique et Science Informatique*, 28, 2009.

- [6] HJ. Heinz. How i lost my owl : Retracting knowledge from el concepts. Master's thesis, Koblenz-Landau University, 2018.
- [7] P. Marquis, O. Papini, and H. Prade, editors. *Panorama de l'intelligence artificielle, ses bases méthodologiques, ses développements*, chapter 13.2. Cépaduès Éditions, 2014.
- [8] T. Di Noia, E. Di Sciascio, and F. M. Donini. Semantic matchmaking as non-monotonic reasoning : A description logic approach. *Journal of Artificial Intelligence Research*, 29(269-307), jul 2007.
- [9] C. Rey. *Découverte des meilleures couvertures d'un concept en utilisant une terminologie Application à la découverte de services web sémantiques*. PhD thesis, Université Clermont II, 2004.
- [10] G. Teege. Making the difference : A subtraction operation for description logics. In *KR'94, Bonn Germany*.

Double adaptation de domaine pour la détection d’objets

Hiba Alqasir, Damien Muselet, Christophe Ducottet¹

¹ Université de Lyon, UJM-Saint-Etienne, CNRS, IOGS, Laboratoire Hubert Curien UMR5516, F-42023, Saint-Etienne, France

Résumé

En apprentissage automatique, l’adaptation de domaine est nécessaire pour améliorer les performances d’un modèle lorsque les distributions des données d’apprentissage et de test sont différentes. Dans ce travail, nous abordons ce problème dans le cadre de la détection d’objets. Après une analyse détaillée du détecteur Faster R-CNN classique, nous montrons que l’adaptation du sous-réseau de proposition de région est cruciale et proposons une solution originale. Nous menons des expériences dans deux contextes d’application différents, la conduite autonome et la vidéo-surveillance des remontées mécaniques, et montrons que notre schéma d’adaptation améliore clairement les résultats de la solution précédente.

Mots-clés

Détection d’objets, adaptation de domaine, apprentissage en profondeur, Faster R-CNN.

Abstract

Domain adaptation is needed to be able to train a model performing well in a target domain with few or no annotations. In this work, we address this domain adaptation problem in the context of object detection. After a detailed analysis of the classical Faster R-CNN detector, we show that adapting the region proposal sub-network is crucial and propose an original way to do it. We run experiments in two different application contexts, namely autonomous driving and ski-lift video surveillance, and show that our adaptation scheme clearly outperforms the previous solution.

Keywords

Object detection, Domain Adaptation, Deep learning, Faster R-CNN.

1 Introduction

Object detection in images refers to the task of automatically finding all instances of given object categories outputting, for each instance, a bounding box and the object category. Recently, approaches based on deep Convolutional Neural Networks (CNNs) have invaded the field thanks to both their efficiency and their outstanding performances [18, 17]. To address a given computer vision problem, these methods require large training datasets with instance-level annotations. However, for most real world applications, few annotations are available due to the lack

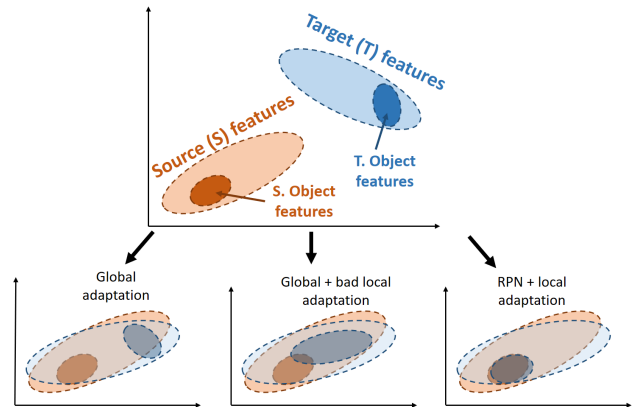


FIGURE 1 – Illustration of global, local and RPN adaptation (see text for details).

of image sources, copyright issues or annotation cost. To overcome this problem, a current trend consists in training the network on a large public annotated dataset (source domain), while adapting the network features to the tested dataset (target domain). This approach is called domain adaptation [14, 23]. If no annotations are available in the target domain, the domain adaptation is referred as unsupervised. In this context, the case of autonomous driving has been extensively addressed and a variety of datasets exists covering different urban scenes situations, illumination and weather conditions [4, 6]. In this paper, we are particularly interested in unsupervised domain adaptation in a ski lift video surveillance scenario. The purpose is to detect dangerous situations during chairlift boarding by detecting relevant objects of the scene (e.g. security bar, skiers). Instance level annotations are available for a reference chairlift where the chair model, the perspective view or boarding system may be different from the target chairlift [2].

Surprisingly, few works explicitly address the problem of unsupervised domain adaptation for object detection. Most approaches study the supervised case basically by fine-tuning a model pre-trained on the source dataset with few annotated images from the target domain, eventually freezing some layers to concentrate the training on the last layers [8, 22]. Other recent approaches try to reduce the domain shift by transforming the source domain to make it close to the target one using style transfer [12]. The most significant contribution of domain adaptive object detection was proposed by [3]. Following [5], they added adversarial

training components in the classical Faster R-CNN detector, in order to adapt both globally and locally the detector. Given the features from the two domains and considering the subset of features specific to the object (Fig. 1), a global adaptation, as illustrated in the bottom left of Fig. 1, may not match source and target object features. Thus, Chen et al. [3] also propose to adapt the features pulled from the regions returned by the Region Proposal Network (RPN). We argue that, since the RPN is trained on the source domain, the proposals from the target images may be wrongly detected and the local features used for the adaptation may be outside the target object features set (bottom center in Fig. 1). In this paper, we propose to adapt the RPN in order to ensure the features extracted from the target images overlap with the source object features. A local adaptation through adversarial learning will thus better align source and domain features (bottom right in Fig. 1).

Our contributions are threefold : 1) We present a new viewpoint about the domain shift problem in object detection. 2) We propose to adapt the RPN as a global feature adaptation and integrate this new adaptation module in Faster R-CNN. 3) We run extensive experiments in two different applications contexts : autonomous driving and ski lift video surveillance.

2 Related Work

Object Detection The first approaches proposed in the context of CNN were based on the region pooling principle [21, 8]. In R-CNN [8], candidate regions detected by selective search were represented by a subset of pooled features and evaluated by an instance classifier. This two-stages principle was further refined in Faster R-CNN [18] with a common CNN backbone to extract the whole image features and two different sub-pipelines : the first one called RPN to generate proposals of regions which are likely to contain objects and the second one which is basically a classification and regression network aiming to refine the location and size of the object and to find its class. Besides these two-stage approaches, one-stage approaches directly predict box location, size and class in a single pipeline either by using anchor boxes with different aspect ratios [13] or by solving a regression problem on the feature grid [17]. Interested readers can refer to the review of recent advances in object detection in [1]. Since Faster R-CNN [18] provides very accurate results and has been largely studied, we propose to consider this network as a baseline in this paper.

Domain Adaptation Unsupervised domain adaptation is needed when we want to learn a predictor in a target domain without any annotated training samples in this domain [14, 23]. Obviously, annotations are available in a source domain which is supposed to be close to the target one. Two main types of methods have been proposed in this context. The first one is to try to match the feature distribution in the source and target domains either by finding a transformation between the domains [15] or by directly adapting the features [10]. One noticeable example

is the gradient reversal layer approach proposed by Ganin et al. [5] that attempts to match source and target feature distributions. They propose to jointly optimize the class predictor and the source-target domain disparity by back-propagation. The second type of methods relies on Generative Adversarial Networks (GANs) [11]. The principle is to generate annotated synthetic target images from the source images and to learn (or fine-tune) the network on these synthetic target data [12].

Domain adaptation for object detection Few works consider domain adaptation for object detection particularly in the unsupervised setting. [16] proposes class-specific subspace alignment to adapt RCNN [8] and [3] uses adversarial training inspired by [5] to adjust features at two different levels of a Faster R-CNN architecture. The adaptation at image level intends to eliminate the domain distribution discrepancy at the output of the backbone network while the instance level adaptation concerns the features which are pooled from a Region of Interest (RoI), before the final category classifiers. Following the same adversarial training approach, Saito et al. [19] argue that a global matching may hurt performance for large domain shifts. They thus propose to combine a strong alignment of local features and a weak alignment of global ones. To the best of our knowledge, none of the previous works considers the adaptation of the region proposal sub network of Faster R-CNN. They are then sensitive to any shift in the distribution of object bounding boxes between source and target domains.

In this work, we propose to incorporate two adversarial domain adaptation modules in Faster R-CNN : the first one at RPN-level to address the source-target domain shift of features of the region proposal module and the second one at instance-level to adapt the RoI-pooled features used in the final classification module.

3 Our approach

In order to explain our adaptation scheme, we have to explain in details the work-flow of Faster R-CNN [7], summarized in Fig. 2. Then, we present our approach to adapt this detector between different domains.

3.1 Faster R-CNN

Faster R-CNN is basically composed of two convolutional blocks called C_1 and C_2 , providing two feature maps F_1 and F_2 , respectively (cf. Fig. 2). Based on F_2 , the RPN predicts a set of box positions used to crop the F_1 feature map using the RoI pooling layer (called RP layer, hereafter). It is worth mentioning that the gradient can not be back-propagated through the RP layer towards the RPN, because this step is not differentiable. The authors of Faster R-CNN resort to an alternating training to cope with this problem [7]. It is crucial to understand this point when one wants to apply domain adaptation to Faster R-CNN. It means that we can not just plug a domain adaptation module after the last layers of Faster R-CNN (namely F_{3i}) and adapt in one shot the classification layers and the convolution blocks C_1

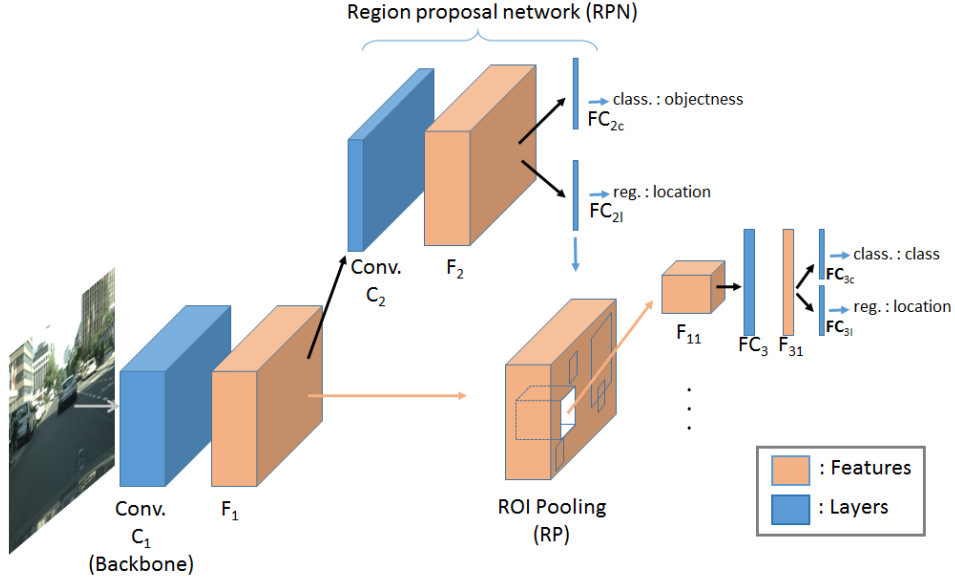


FIGURE 2 – Faster R-CNN Workflow.

and C_2 .

Back to the workflow of Faster R-CNN, the outputs F_{1i} , $i = 1, \dots, N_p$, of the RP layer are cropped and resized parts of the F_1 feature map. N_p is the number of proposals returned by the RPN. The feature maps F_{1i} are then sent to shared fully connected layers FC_3 whose outputs F_{3i} are used to take the final decision of class and location.

From this workflow, we note that the classification and regression layers take as inputs either F_2 or F_{3i} , which are the key feature maps of the detector. In the next section, we present how these feature maps can be adapted between the two domains.

3.2 Adapting Faster R-CNN

Let us consider a source domain \mathcal{S} with N_S images $\{I_i^S\}$, $i = 1, \dots, N_S$, each containing n_i^S objects, located at the positions l_{ij}^S and associated with the classes c_{ij}^S , $j = 1, \dots, n_i^S$. Likewise, we denote \mathcal{T} a target domain constituted of N_T target images $\{I_i^T\}$, $i = 1, \dots, N_T$, each containing n_i^T objects, located at the positions l_{ij}^T and associated with the classes c_{ij}^T , $j = 1, \dots, n_i^T$.

If the two domains are different (cameras, viewpoints, weather conditions, ...), there exists a domain shift between the joint distributions $P(I^S, l^S, c^S)$ and $P(I^T, l^T, c^T)$. In this case, we can not train the detector on the source data and obtain good results on the target data, without adaptation. The aim of domain adaptation is to decrease this distribution discrepancy so that $P(I^S, l^S, c^S) \approx P(I^T, l^T, c^T)$. In the context of unsupervised domain adaptation, the labels (locations and classes) of the target data are not available and this is not an easy task to decrease the joint distribution discrepancy. By applying the Bayes' rule on the joint distribution, we obtain, for the source domain :

$$P(I^S, l^S, c^S) = P(l^S, c^S | I^S) P(I^S) \quad (1)$$

Most of the domain adaptation approaches assume a covariate shift, which means that the shift between the source and target joint distributions is caused by the marginal distributions $P(I)$, while the conditional distributions $P(l, c | I)$ are constant across domains, i.e. $P(l^S, c^S | I^S) = P(l^T, c^T | I^T)$. Under this assumption, in order to decrease the joint distribution discrepancy, we have just to decrease the marginal distribution shift, so that $P(I^S) \approx P(I^T)$. In order to change the marginal distributions of the images, the classical approaches apply a transform T on the image features, so that $P(T(I^S)) \approx P(T(I^T))$. Usually, the transform T is a part of a convolution neural network.

In this paper, we propose to consider and adapt different feature maps extracted from the images. By looking at Fig. 2, we note that two feature maps are used as input for classification and regression layers, namely the F_2 feature map and the F_{3i} feature vectors. So, in order to adapt the detector to the source domain, we have to adapt the marginal distributions of F_2 and F_{3i} , so that $P(F_2^S) \approx P(F_2^T)$ and $P(F_{3i}^S) \approx P(F_{3i}^T)$.

In order to enforce these distributions to be closer, we propose to resort to an adversarial domain adaptation approach [5] called GRL for gradient reversal layer. Note that any other adversarial domain adaptation algorithms could have been used, we just use this one for a fair comparison with DA-Faster [3]. When plugged on a feature map F_k , the idea of GRL is to minimize the discrepancy between the feature distributions over the source and target domains $P(F_k^S)$ and $P(F_k^T)$ [5]. If the GRL is able to perfectly overlap these two distributions, we can conclude that the features extracted at this point of the network (F_k) are domain invariant and so can be applied either on the source or target domain with equivalent accuracies.

From the previous analysis, it is obvious that two GRL modules should be inserted in the detector : one after the

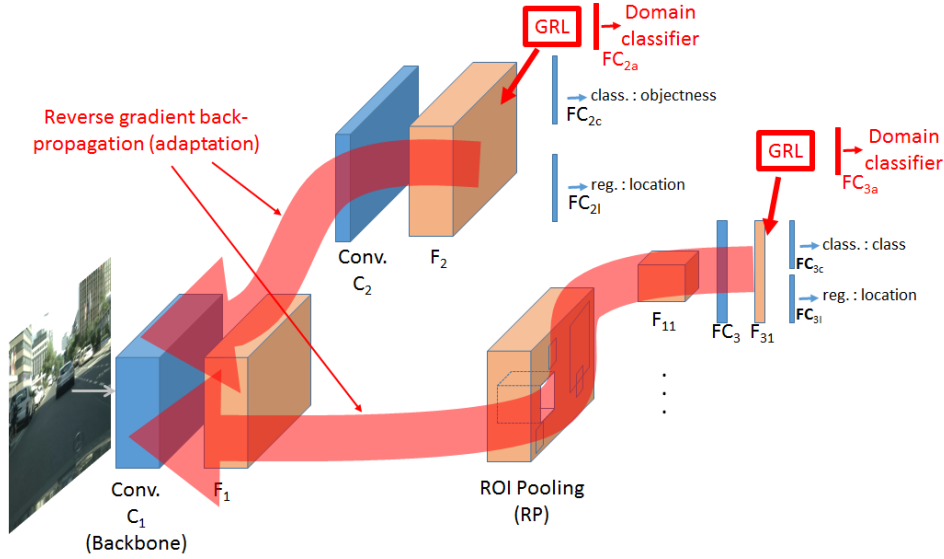


FIGURE 3 – Our domain adaptation for Faster R-CNN. See text for details

feature map F_2 and one after the feature vector F_{3i} . It is worth mentioning that, when we plug a GRL module to a feature map, we back-propagate the (reverse-)gradient until the first layer of the C_1 convolutional block. Thus, the main advantage of our approach is that the reversal gradients are back-propagated through all the layers of the detector. Consequently, the backbone, the RPN and the local features are all adapted (see Fig. 3).

Formally, at training time, the total loss corresponding to a given training image $I_k \in I^S \cup I^T$ from domain $d_k \in \{S, T\}$ is given by :

$$L = L_{Fst} - \lambda \sum_{i,j} L_H \left(FC_{2a}(F_2^{i,j}(I_k)), d_k \right) - \lambda \sum_{i=1}^{N_p} L_H (FC_{3a}(F_{3i}(I_k)), d_k) \quad (2)$$

where L_{Fst} denotes the original Faster R-CNN loss activated only if $I_k \in I^S$, L_H denotes the cross-entropy loss, λ denotes the trade-off parameter to balance Faster R-CNN loss and domain adaptation losses, FC_{2a} and FC_{3a} denote the fully connected predictors for domain adaptation, $F_2^{i,j}(I_k)$ denotes the feature vector at location (i, j) of feature map F_2 for image I_k , and $F_{3i}(I_k)$ denotes the feature vector corresponding to the proposal region i of image I_k . We note that the recent domain adaptive detection approaches ([3, 19]) have not tried to adapt the RPN layer and we think that this is a strong weakness of these approaches. Indeed, as mentioned in [3] (called DA-Faster hereafter), the image-level adaptation is enforcing the F_1 target and source feature distributions to be closer but it is very hard to perfectly align them. This is one of the reasons why DA-Faster approach also applies instance level adaptation. But, it is clear in Fig. 2, that if F_1 features are not well adapted between the domains, the output of the RPN will also be

different between the domains and consequently, the locations where the boxes F_{1i} are cropped from F_1 will be domain dependent. Therefore, the instance-level adaptation on F_{3i} features will not help to adapt the object detector between domains, since it will work on local features which are not equivalent between the domains (see Fig. 1).

4 Experiments

4.1 Experiment Setup

To evaluate the efficiency of our approach we conduct experiments in two contexts : autonomous driving and video surveillance of ski lifts. In each case, we train on a source dataset and test on a target dataset from a different domain. During training, likewise the other domain adaptive approaches, we use also images from the target domain, but without any label, while the source dataset images are provided with their bounding boxes instance annotations.

The baseline is Faster R-CNN model trained only on the source dataset. As mentioned earlier, our solution is inspired from DA-Faster [3] but our contribution is in the analysis of the domain shift in Faster R-CNN, conducting to the solution that the domain adaptation module (GRL) should be plugged at the RPN level. Consequently, the aim of these experiments is to compare DA-Faster with our approach in order to check the validity of our contribution in practice. Thus, adapting the RPN in other solutions such as [19], or using other adaptation modules than GRL such as [14, 23] might have provided better results, but it is out of the scope of this paper. Thus, for all the experiments, we compare our approach with DA-Faster [3].

As mentioned in [19], the results provided by the authors of DA-Faster are unstable and Saito et al. proposed to re-implement their own code for DA-Faster, conducting to lower results than the original paper [3]. So likewise [19], we report the results of DA-Faster with the implementa-

TABLE 1 – Detection results on Foggy Cityscapes (trained on Cityscapes dataset). The AP50 is reported for each class as well as the average APcoco, AP50 and AP75 over all classes.

| | person | rider | car | truck | bus | train | mcycle | bicycle | APcoco | AP50 | AP75 |
|------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|--------------|
| Faster R-CNN | 18.8 | 20.5 | 24.2 | 17.0 | 8.0 | 6.2 | 7.2 | 5.0 | 06.20 | 13.35 | 05.42 |
| DA Faster R-CNN | 27.3 | 35.7 | 44.1 | 20.3 | 35.2 | 8.9 | 16.2 | 23.6 | 12.28 | 26.41 | 10.02 |
| DA Faster R-CNN* | 25.0 | 31.0 | 40.5 | 22.1 | 35.3 | 20.2 | 20.0 | 27.1 | - | 27.6 | - |
| Ours | 27.8 | 35.8 | 45.1 | 23.5 | 42.1 | 26.1 | 18.0 | 27.6 | 13.70 | 30.47 | 11.04 |

tion provided by [9] with the same hyper-parameters as our solution (results denoted *DA-Faster* hereafter), as well as the results provided by the original paper [3] (denoted *DA-Faster**), when available on the considered dataset.

To evaluate object detection we report the mean Average Precision (mAP) with intersection over union (IoU) threshold at 0.5 (denoted AP50), the mAP with IoU threshold of 0.75 (AP75) and the mAP averaged over multiple IoU from 0.5 to 0.95 with a step size of 0.05 (APcoco). The network is trained in an end-to-end manner using back-propagation and the stochastic gradient descent (SGD) algorithm. As a standard practice, Faster R-CNN backbone is initialized with pre-trained weights on ImageNet classification. We use a learning rate of 0.001 for 50k iterations, and 0.0001 for the next 20k iterations. Each iteration has 2 mini-batches, one from source domain and the other from target domain. The trade-off parameter λ to balance Faster R-CNN loss and domain adaptation loss is set to 0.1 as in [3]. We use a momentum of 0.9 and a weight decay of 0.0005.

4.2 Autonomous driving

In this context we evaluate the domain adaptive detectors for two domain shifts : weather conditions (foggy and not foggy) and acquisition conditions (different cameras, different viewpoints and different scenes).



FIGURE 4 – One image from each dataset : the Cityscapes dataset (left), its foggy version (right) and the KITTI dataset (bottom).

Cityscapes \rightarrow Foggy Cityscapes. In the first experiment we use the Cityscapes dataset [4] as source domain. It is a urban scene dataset with 2975 training images and 500 validation images. The 1525 unlabeled images are not considered. For training the network, we are using the 2975 train

images and do not consider the validation images. There are 8 categories with instance annotations in this dataset, namely *person*, *rider*, *car*, *truck*, *bus*, *train*, *mortorcycle* and *bicycle*. The target domain is the Foggy Cityscapes [20] dataset generated by applying fog synthesis on the Cityscapes dataset to simulate fog on real scenes (see Fig. 4). Thus, the number of images and labels are exactly the same as for Cityscapes dataset. For testing the detection, we are using the 500 validation images from Foggy Cityscapes. The results are summarized in Table 1. First, we can note that, without domain adaptation, the results of Faster R-CNN are very bad, underlying the strong need of adapting the network in case of weather condition variations. Thus, DA-Faster improves the results over Faster R-CNN, but we note that our approach clearly outperforms DA-Faster on this dataset, showing that the RPN adaptation helps in adapting the detector in case of weather condition variations.

Cityscapes \rightarrow KITTI. In this experiment Cityscapes is the source domain, and KITTI [6] is the target domain (see Fig. 4). KITTI is a benchmark for autonomous driving which consists of 7481 training images. Since the test set is not annotated we use all the training images with their annotations at test time to evaluate the performance. Only one category (*car*) is annotated in KITTI, so we consider this single class for evaluation. The results are summarized in Table 2. Once again, we note that the domain adaptation helps improving Faster R-CNN results. We see also that our approach outperforms DA-Faster for all the criteria when using the same hyper-parameters. The results provided in [3] are better than ours for *AP50*, but note that the implementation and hyper parameters are different from our tests. The comparison is therefore not fair.

TABLE 2 – Detection results in KITTI training set (trained in Cityscapes dataset) for one class (Car) detection.

| | APcoco | AP50 | AP75 |
|------------------|--------------|-------------|--------------|
| Faster R-CNN | 26.73 | 58.60 | 21.54 |
| DA Faster R-CNN | 27.51 | 60.38 | 22.67 |
| DA Faster R-CNN* | - | 64.1 | - |
| Ours | 28.39 | 61.32 | 23.59 |

4.3 Video surveillance of ski lifts

The MIVAO research project was launched in collaboration with a french start-up Bluecime, based on the needs of ski lift operators to secure chairlifts. MIVAO aims to develop a computer vision system that acquires images from

TABLE 3 – Detection results on the chairlift dataset. First, adaptation from chairlift 1 to chairlift 2, and second adaptation from chairlift 2 to chairlift 1.

| | <i>ch1 → ch2</i> | | | <i>ch2 → ch1</i> | | |
|-----------------|------------------|--------------|-------------|------------------|--------------|-------------|
| | APcoco | AP50 | AP75 | AP | AP50 | AP75 |
| Faster R-CNN | 30.34 | 99.49 | 0.30 | 36.56 | 98.98 | 9.86 |
| DA Faster R-CNN | 50.51 | 99.50 | 33.4 | 42.56 | 98.99 | 11.1 |
| Ours | 50.93 | 99.99 | 30.7 | 48.83 | 99.00 | 45.6 |

the boarding station of chairlifts, analyzes the important elements (people, chairlift carrier, safety bar, ...) and triggers an alarm in case of dangerous situations. In this paper, we tackle this problem as an object detection task trying to detect the safety bar in the image, considering that it has to be closed when the chairlift leaves the boarding station. Across the ski resorts, the viewpoint, the background, the carrier geometry and the camera may be different and domain adaptive detectors are required to install new systems without a fastidious and time-consuming step of manual annotation.

Chairlift dataset For this experiment, we have created a dataset with images from two different chairlifts, called hereafter chairlift 1 and chairlift 2. The dataset contains 3864 images from chairlift 1 and 4260 images from chairlift 2. Example images are provided in Fig. 5. We can note that the main differences between the two chairlifts are in the viewpoints which are slightly different and in the presence of a cluttered background in the chairlift 2.



FIGURE 5 – Example images from our chairlift dataset. The two left images are from chairlift 1 and the two right images are from chairlift 2. The box annotations (open :red and close :green) are provided for the two right images, for illustration.

The images are centered on the chairlift and manually labeled with the position and the dimensions of the bounding box containing the safety bar. From this information, we have created instance annotations with two categories : open safety bar and close safety bar, as illustrated on the two right images from Fig. 5.

Evaluation The results are provided in Table 3. By training the baseline Faster R-CNN using images from one chairlift and test it on images from another chairlift, the results were surprisingly very good in terms of AP50. This can be explained by the important size of the ground truth bounding boxes that have high chance to well overlap random bounding boxes with similar dimensions. Obviously, when looking at the more demanding criteria such as *APcoco* or *AP75*, the need of domain adaptation is evident for precise object detection. The results show that the two domain adaptive detectors (DA-Faster and ours) are equivalent for the adaptation from chairlift 1 to chairlift 2, but they also show that our adaptation is much better than DA-Faster for the adaptation from chairlift 2 to chairlift 1. It is difficult to explain why DA-Faster is less accurate in one direction (*ch2 → ch1*) than in the other direction (*ch1 → ch2*). One assumption could be that in DA-Faster, the RPN is better trained on chairlift 1 since in this case the background is less cluttered. Thus, when applying it on chairlift 2, the adaptation process tends to promote features from the foreground and both the proposal and the classification are good. On the contrary, if the RPN is trained on chairlift 2, it will rely on cluttered features which are removed with the global adaptation and thus, for DA-Faster, the proposals will be bad on chairlift 1, leading to an important residual domain shift in the results. On the contrary, in our method, since the RPN is directly adapted, the residual shift is lower (see figure 1 and the related explanation in section 1).

5 Conclusion

In this paper, we have tackled the problem of domain adaptation for object detection. After a detailed analysis of the complete workflow of the classical Faster R-CNN detector, we have proposed to adapt the features pulled from this network at two different levels : one adaptation at a global level in the Region Proposal Network and one adaptation at the local level for each bounding box returned by the RPN. We have shown that these two adaptations are complementary and provide very good detection results. We have tested our solution on two different applications, namely the autonomous driving and the chairlift security. As future works, we propose to test more accurate adaptation procedures such as the approaches presented in [14, 23]. These methods could help in the learning step to reach stable solutions which is a strong weakness of the domain adaptive Faster R-CNN. Furthermore, it could be interesting to adapt the features at different depth of the network as recommended by [19].

Références

- [1] Shivang Agarwal, Jean Ogier Du Terrail, and Frédéric Jurie. Recent advances in object detection in the age of deep convolutional neural networks. *arXiv preprint arXiv :1809.03193*, 2018.
- [2] Kevin Bascol, Rémi Emonet, Elisa Fromont, and Raluca Debuschere. Improving chairlift security with deep learning. In *International Symposium on Intelligent Data Analysis*, pages 1–13. Springer, 2017.
- [3] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3339–3348, 2018.
- [4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.
- [5] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv :1409.7495*, 2014.
- [6] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics : The kitti dataset. *The International Journal of Robotics Research*, 32(11) :1231–1237, 2013.
- [7] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
- [8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [9] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. <https://github.com/facebookresearch/detectron>, 2018.
- [10] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification : A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520, 2011.
- [11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [12] Naoto Inoue, Ryosuke Furuta, Toshihiko Yamasaki, and Kiyoharu Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5001–5009, 2018.
- [13] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd : Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.
- [14] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650, 2018.
- [15] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2) :199–210, 2010.
- [16] Anant Raj, Vinay P Namboodiri, and Tinne Tuytelaars. Subspace alignment based domain adaptation for rnn detector. *arXiv preprint arXiv :1507.05578*, 2015.
- [17] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once : Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn : Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
- [19] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019.
- [20] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, pages 1–20, 2018.
- [21] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat : Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv :1312.6229*, 2013.
- [22] Bharat Singh and Larry S Davis. An analysis of scale invariance in object detection snip. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3578–3587, 2018.
- [23] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

Méthode d'analyse sémantique d'images combinant apprentissage profond et relations structurelles par appariement de graphes

J. Chopin^{1,2}, J.-B. Fasquel¹, H. Mouchère², I. Bloch³, R. Dahyot⁴

¹ LARIS, Université d'Angers, Angers, France

² LS2N, Université de Nantes, France

³ LTCI, Télécom Paris, Institut Polytechnique de Paris, France

⁴ School of Computer Science & Statistics, Trinity College Dublin, Ireland

Résumé

Nous proposons une méthode de segmentation sémantique d'images, combinant apprentissage profond et relations spatiales entre régions. Cette méthode repose sur l'appariement inexact de graphes, appliqué en sortie d'un réseau de neurones profond. Notre proposition est évaluée sur une base publique dédiée à la segmentation de visages, en mesurant l'IoU ("Intersection over Union") des boîtes englobantes des régions obtenues avec et sans utilisation des relations spatiales. Celles-ci permettent une amélioration de 2% en moyenne, et jusqu'à 24% dans certains cas.

Mots-clés

Vision par ordinateur, Apprentissage profond, Appariement inexact de graphes, Problème d'affectation quadratique.

Abstract

We propose a method for semantic image segmentation, combining a deep neural network and spatial relationships between image regions, encoded in a graph representation of the scene. Our proposal is based on inexact graph matching, applied to the output of a deep neural network. The proposed method is evaluated on a public dataset used for segmentation of images of faces. Preliminary results show that, in terms of IoU of region bounding boxes, the use of spatial relationships lead to an improvement of 2.4% in average, and up to 24.4% for some regions.

Keywords

Computer vision, Deep learning, Inexact graph matching, Quadratic assignment problem.

1 Introduction

L'apprentissage profond a montré son efficacité dans de nombreux domaines [8], en particulier pour la segmentation sémantique d'images en vision par ordinateur [7]. L'une des limites des approches par réseaux de neurones profonds est la nécessité de disposer d'un ensemble de données d'apprentissage vaste, représentatif et annoté [16]. De plus, la plupart de ces approches n'utilisent que les images elles-mêmes (les données), sans connaissance a priori sur les structures qu'elles contiennent et leur agence-

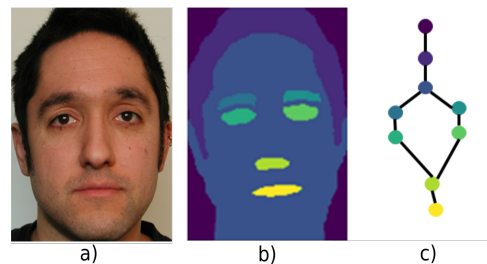


FIGURE 1 – Segmentation sémantique et relations spatiales. Les images sont extraites de la base de données publique FASSEG [10, 11]. a) Image initiale. b) Segmentation sémantique où chaque région appartient à une classe spécifique (par exemple l'oeil gauche, l'oeil droit). c) Relations spatiales modélisées par un graphe où chaque sommet correspond à une région spécifique de b : fond, cheveux, sourcil gauche/droit, oeil gauche/droit, nez, bouche. Les arêtes portent des relations correspondant aux distances entre les régions dans notre cas. Par souci de clarté, seules certaines arêtes du graphe complet sont affichées.

ment spatial. De plus, ces méthodes nécessitent d'optimiser un grand nombre de paramètres.

Ce type d'approche ignore les informations structurelles observables à haut-niveau. Cela peut concerner les relations spatiales entre différentes entités, comme l'illustre la figure 1 avec les positions relatives entre les principales régions du visage observées dans l'image annotée. Bien que souvent ignorées, ces informations structurelles ont montré leur potentiel dans de multiples travaux connexes, considérant des relations spatiales, d'inclusion ou de photométrie [1, 4, 5, 15], souvent appliquées à l'imagerie médicale [2, 3, 14]. Ces informations sont généralement représentées à l'aide de graphes, où les sommets correspondent à des régions et les arêtes portent les informations structurelles. L'identification des régions peut alors être réalisée par appariement de graphes [4, 5, 12], par raisonnement séquentiel dans les graphes, ou global par satisfaction de contraintes.

Dans ce contexte, nous proposons de combiner deux approches : les réseaux de neurones profonds qui s'avèrent

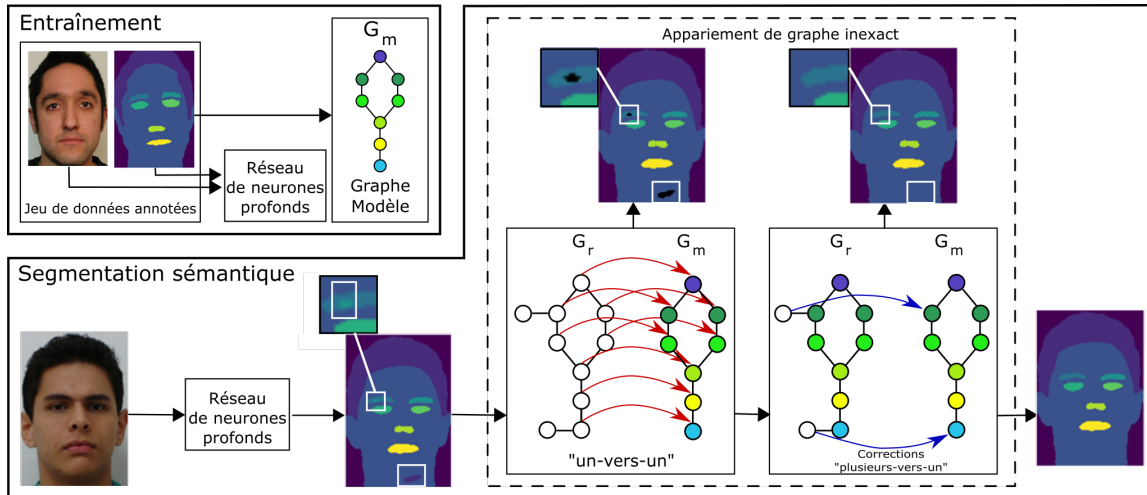


FIGURE 2 – Aperçu de la méthode. **Entraînement** : l'ensemble des données annotées est utilisé pour entraîner le réseau de neurones et construire un graphe modèle (similaire à celui de la figure 1-c). Par souci de clarté, bien que les graphes soient complets, seules quelques arêtes sont indiquées. La couleur des sommets correspond à la couleur des régions associées. **Segmentation sémantique** : le réseau de neurones produit une segmentation sémantique, éventuellement avec des artefacts (par exemple la région claire à l'intérieur du sourcil et la région sombre sur le cou). Un graphe G_r est ensuite construit à partir de cette segmentation et mis en correspondance avec le graphe modèle G_m . Cet appariement inexact entre graphes est réalisé en deux étapes : 1. les régions correctement segmentées sont récupérées (appariement un-vers-un), les artefacts étant ignorés (deux sommets restants dans cet exemple) ; 2. les artefacts restants sont mis en correspondance (appariement plusieurs-vers-un). Dans cet exemple, on peut voir que les deux artefacts sont correctement ré-étiquetés (voir les zones entourées).

efficaces mais nécessitent souvent de grands ensembles de données d'entraînement, et les graphes pour encoder des relations structurelles de haut niveau au sein d'images. Il est à noter que l'utilisation des réseaux de neurones profonds pour l'appariement de graphes est un sujet qui suscite actuellement beaucoup d'intérêt dans la communauté scientifique, pour des problèmes autres que la vision par ordinateur, par exemple en biologie, en sciences sociales, en linguistique [6, 9, 13]. A noter que ces travaux intègrent l'appariement de graphes dans le réseau de neurones profond, tandis que, dans notre cas, l'appariement est appliqué sur la sortie du réseau de neurones.

L'originalité de notre proposition concerne la formulation de la combinaison entre un réseau de neurones profond et l'appariement de graphes exploitant les relations spatiales, appliqué en sortie du réseau. Elle permet de corriger la segmentation sémantique, obtenue par l'utilisation de la carte de probabilité en sortie du réseau de neurones, en prenant en compte les relations spatiales observées dans la base de données annotées. L'utilisation de la structure spatiale globale de la scène permet également d'être moins sensible à la diversité (et donc à la taille) du jeu de données d'entraînement utilisé par le réseau de neurones.

La méthode proposée est détaillée dans la section 2. Des expériences préliminaires illustrant le potentiel de cette proposition sont présentées dans la section 3. La section 4 conclut le document par une discussion.

2 Apprentissage profond et connaissances structurelles

La figure 2 donne une vue d'ensemble de la méthode proposée avec les images considérées dans les expériences. À l'aide d'un ensemble de données d'apprentissage annoté, le réseau de neurones profond est entraîné à effectuer une segmentation sémantique. De plus, en utilisant uniquement les images annotées, les relations spatiales entre les différentes régions sont mesurées (telles que la moyenne des distances), ce qui conduit à un graphe modèle G_m où les sommets et les arêtes correspondent respectivement aux régions annotées et aux relations spatiales.

Lors du traitement d'une image inconnue, le réseau de neurones fournit une segmentation, à partir de laquelle un graphe d'hypothèse G_r est construit. Ce graphe est ensuite mis en correspondance avec le graphe modèle. L'objectif est de faire correspondre les sommets (et donc les régions sous-jacentes) produits par le réseau de neurones avec ceux du modèle, ce qui implique le ré-étiquetage de certaines régions (cf. appariement plusieurs-vers-un dans la figure 2). Cela produit une segmentation sémantique finale correspondant aux relations de haut niveau observées dans l'ensemble des données d'entraînement.

Nous détaillons ci-après l'étape de construction du graphe hypothèse à partir du réseau de neurones profond (Section 2.1) puis son appariement avec le graphe modèle (Section 2.2).

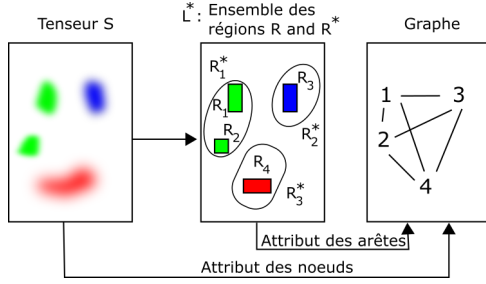


FIGURE 3 – Construction d'un graphe à partir du tenseur S et de la segmentation \mathcal{L}^* qui en résulte. Chaque point de l'image de gauche est associé à un vecteur de probabilité représenté par des dégradés de couleurs. R_1^* est l'ensemble des régions (R_1 et R_2) qui appartiennent à la classe 1. Les attributs des arêtes sont calculés à partir des relations spatiales entre les régions R_i . Les attributs des sommets sont des vecteurs de probabilité moyens, calculés sur des régions R_i associées.

2.1 Construction des graphes

L'image d'entrée est traitée par le réseau de neurones qui produit, en sortie, un tenseur $S \in \mathbb{R}^{I \times J \times C}$ avec I la largeur (en pixels) de l'image, J la hauteur (en pixels) de l'image et C le nombre total de classes.

À l'emplacement du pixel (i, j) , la valeur $S(i, j, c) \in [0, 1]$ est la probabilité d'appartenir à la classe c considérée dans la segmentation, avec les contraintes :

$$(\forall c = 1, \dots, C, 0 \leq S(i, j, c) \leq 1) \wedge \left(\sum_{c=1}^C S(i, j, c) = 1 \right)$$

La carte de segmentation \mathcal{L}^* sélectionne l'étiquette c de la classe ayant la plus forte probabilité :

$$\forall (i, j) \in \{1, \dots, I\} \times \{1, \dots, J\}, \\ \mathcal{L}^*(i, j) = \arg \max_{c \in \{1, \dots, C\}} S(i, j, c)$$

À partir de cette carte de segmentation, nous définissons l'ensemble R de toutes les composantes connexes résultantes (voir figure 3, où $R = \{R_1, \dots, R_4\}$). Nous définissons également un ensemble $R^* = \{R_1^*, \dots, R_C^*\}$, où, pour chaque classe $c \in \{1, \dots, C\}$, R_c^* est l'ensemble de régions correspondant aux composantes connexes appartenant à la classe c selon le réseau de neurones (voir figure 3, où $R^* = \{R_1^*, \dots, R_3^*\}$). Cet ensemble R^* est utilisé pour contraindre l'appariement des graphes comme décrit dans la section 2.2.1.

À partir de l'ensemble R , une représentation structurelle est construite et modélisée par le graphe $G_r = (V_r, E_r, A, D)$, où V_r est l'ensemble des sommets, E_r l'ensemble des arêtes, A un interpréteur de sommets et D un interpréteur d'arêtes. Chaque sommet $v \in V_r$ est associé à une région $R_v \in R$ avec un attribut, fourni par la fonction A , qui est le vecteur de probabilité d'appartenance moyen sur l'ensemble des pixels $p = (i, j)$ composant R_v ,

donc calculé sur le tenseur S initial (voir figure 3) :

$$\forall v \in V_r, c \in \{1, \dots, C\}, A(v)[c] = \frac{1}{|R_v|} \sum_{(i,j) \in R_v} S(i, j, c) \quad (1)$$

Nous considérons un graphe complet où chaque arête $e = (a, b) \in E_r$ a un attribut défini par la fonction D , associé à une relation entre les régions R_a et R_b (cf. figure 3). Dans notre cas, nous choisissons la distance minimale entre les deux régions :

$$\forall e = (a, b) \in E_r, D(e) = \min_{p \in R_a, q \in R_b} (\|p - q\|) \quad (2)$$

Le graphe modèle $G_m = (V_m, E_m, A, D)$, composé de C sommets (un sommet par classe), est construit à partir de l'ensemble d'entraînement. L'attribut d'un sommet sera un vecteur de dimension N avec une seule composante non nulle (de valeur égale à 1), associée à l'indice de la classe correspondante. Les arêtes E_m sont calculées en agréant les occurrences de relations entre les régions annotées des images d'entraînement. Dans cet article, la relation considérée est la distance minimale entre régions et l'agrégation est faite par leur moyenne.

2.2 Appariement avec le graphe modèle

Afin d'identifier les régions, l'objectif est d'associer chacun des sommets de G_r à un sommet du graphe modèle G_m . Selon l'hypothèse réaliste d'avoir plus de régions dans l'image associée à G_r que dans le modèle (c'est-à-dire $|V_r| \geq |V_m|$), nous sommes confrontés à un problème d'appariement inexact de graphes de type plusieurs-vers-un [12]. Nous proposons de formuler cet appariement comme un problème d'affectation quadratique (QAP, pour *Quadratic Assignment Problem*), comme cela a été récemment considéré [17].

Dans notre cas, la mise en correspondance est représentée par une matrice $X \in \{0, 1\}^{|V_r| \times |V_m|}$, où $X_{ij} = 1$ signifie que le sommet $i \in V_r$ est mis en correspondance avec le sommet $j \in V_m$. Cela est illustré dans la figure 4 dans deux cas (correspondances "un-vers-un" et "plusieurs-vers-un"). L'objectif est de déterminer la meilleure correspondance (X^*), solution de :

$$X^* = \arg \min_X \{ \text{vec}(X)^T K \text{vec}(X) \} \quad (3)$$

où $\text{vec}(X)$ est la représentation sous forme de vecteur colonne de la matrice X et T est l'opérateur de transposition. La matrice K , non détaillée ici par souci de concision (voir [17] pour plus de détails), intègre les mesures de dissimilarité entre les deux graphes G_r et G_m , au niveau des sommets (éléments diagonaux) et des arêtes (éléments non diagonaux) :

$$K = \alpha K_v + (1 - \alpha) \frac{K_e}{\max K_e} \quad (4)$$

où K_v intègre les dissimilarités entre les sommets (distance euclidienne entre les vecteurs de probabilité d'appartenance à une classe), et $\alpha \in [0, 1]$ un paramètre. Dans

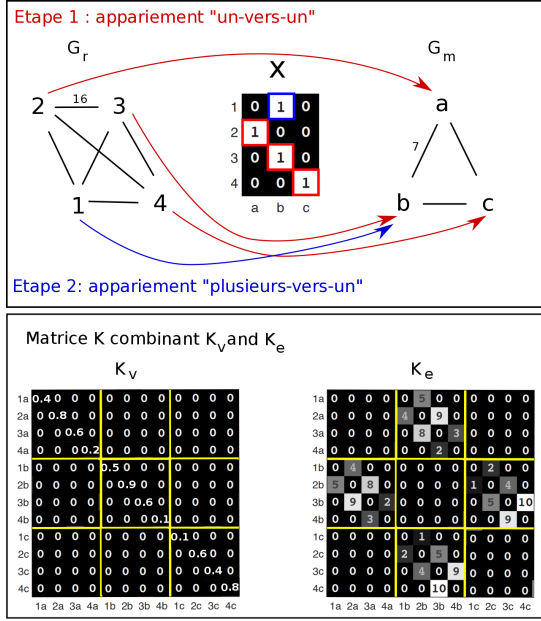


FIGURE 4 – Appariement de graphes formulé comme un QAP (illustration inspirée de [17]), où la matrice X matérialise un appariement entre les graphes G_r et G_m . La première étape (en rouge) vise à chercher une correspondance de type "un-vers-un" (chaque sommet de V_m est associé à un seul sommet de V_r). La deuxième étape (en bleu) vise à faire correspondre les sommets restants de G_r , ce qui conduit à un appariement final de type "plusieurs-vers-un". Pour trouver l'appariement optimal, la matrice K est utilisée, combinant les matrices K_v et K_e , qui mesurent respectivement les dissimilarités entre les sommets et les arêtes. Par souci de clarté, seuls les attributs scalaires de deux arêtes sont indiqués.

l'exemple considéré dans la figure 4, $K_v[1, 1] = 0, 4$ (ligne et colonne nommées 2a) représente la dissimilarité, en termes de vecteurs de probabilité, entre les sommets 2 de G_r et a de G_m , si l'on faisait correspondre ces deux sommets.

La matrice K_e est liée aux dissimilarités entre les arêtes. Par exemple, dans la figure 4, $K_e[6, 1] = 9$ (ligne et colonne respectivement nommées 3b et 2a) correspond à la dissimilarité entre les arêtes $(2, 3) \in E_r$ (attribut scalaire valant 16) et $(a, b) \in E_m$ (attribut scalaire valant 7), si nous faisons correspondre simultanément le sommet 2 avec le sommet a et le sommet 3 avec le sommet b. Dans un tel cas, $K_e[6, 1]$ est calculé en utilisant ces deux attributs : $K_e[6, 1] = 16 - 7 = 9$. Les termes K_e sont liés aux distances entre les régions (normalisées dans la matrice finale K).

Le paramètre α ($\alpha \in [0, 1]$) permet de pondérer la contribution relative des dissimilarités entre les sommets et les arêtes (les termes K_v varient entre 0 et 1, et K_e est normalisé selon l'équation 4).

En raison de la nature combinatoire de ce problème d'optimisation [17], nous proposons une procédure en deux

étapes, s'appuyant sur la segmentation sémantique initiale fournie par le réseau de neurones, et consistant à :

1. rechercher une première correspondance "un-vers-un" (cf. figure 4- étape 1);
2. affiner l'appariement par la mise correspondance des sommets restants, conduisant finalement à une correspondance "plusieurs-vers-un" (cf. figure 4- étape 2).

2.2.1 Appariement initial : "un-vers-un"

On cherche la solution optimale à l'équation 3 en imposant les trois contraintes suivantes à X , réduisant ainsi l'espace de recherche pour les candidats éligibles :

1. $\sum_{j=1}^{|V_m|} X_{ij} \leq 1$: certains sommets i de G_r peuvent n'être associés à aucun sommet de G_m .
2. $\sum_{i=1}^{|V_r|} X_{ij} = 1$: chaque sommet j de G_m doit être associé à un seul sommet de G_r (cas des 1 entourés en rouge dans la figure 4).
3. $X_{ij} = 1 \Rightarrow R_i \in R_j^*$: le sommet $i \in V_r$ peut être mis en correspondance avec le sommet $j \in V_m$ si la région R_i associée a été initialement considérée par le réseau de neurones comme appartenant très probablement à la classe j ($R_i \in R_j^*$). Par exemple, dans le cas de la figure 3, seuls les sommets liés aux régions R_1 et R_2 seraient considérés comme candidats pour la classe 1 (R_1^*).

Les deux premières contraintes 1 et 2 assurent la recherche d'un appariement de type "un-vers-un". Grâce à la troisième contrainte, on réduit l'espace de recherche en s'appuyant sur le réseau de neurones : on suppose qu'il a correctement, au moins dans une certaine mesure, identifié les régions cibles, même si des artefacts peuvent encore avoir été produits (à gérer en affinant ultérieurement l'appariement). Cette étape nous permet de retrouver la structure générale des régions (modélisée par G_m).

2.2.2 Appariement final : "plusieurs-vers-un"

Nous associons chaque sommet k restant ($k \in V_r \mid \sum_{j=1}^{|V_m|} X_{kj} = 0$) au sommet $i^* \in V_r$ en considérant la fonction de coût suivante entre deux sommets i et k de G_r :

$$\text{cost}(i, k) = \alpha |A(k) - A(i)| + (1 - \alpha) \frac{D(k) - D(i)}{\max_{u \in E_r} D(u)}. \quad (5)$$

$$i^* = \underset{i \in V_r \mid \sum_{k=1}^{|V_m|} X_{ik} = 1}{\text{argmin}} \text{cost}(i, k). \quad (6)$$

Selon cette formulation, il apparaît que les sommets restants sont appariés aux sommets de G_m en recherchant indirectement des correspondances avec les sommets déjà appariés de G_r . Par conséquent, on se concentre sur les similarités au sein de l'image actuelle et non avec le modèle. Dans la figure 4, cela correspond à trouver la correspondance entre les sommets 1 et b en étudiant indirectement la pertinence de la correspondance des sommets 1 et 3 (le sommet 3 étant déjà apparié avec le sommet b).

La formulation de l'équation 6 et concernant un seul sommet est similaire à la formulation matricielle des équations 3 et 4 (et concernant simultanément plusieurs sommets). La seule différence est que nous considérons les dissimilarités des sommets et des arêtes dans le graphe G_r au lieu de considérer les dissimilarités entre G_r et G_m .

3 Expérimentations

Nous présentons, ci-après, l'ensemble de données considéré, puis le protocole d'évaluation, et enfin les résultats.

3.1 Données

Nous considérons le jeu de données public FASSEG¹. Celui-ci porte sur la segmentation sémantique multi-classes du visage [10] (cf. figure 1) ainsi que l'estimation de sa pose [11]. Pour cette étude préliminaire, nous considérons un sous-ensemble de ce jeu de données qui correspond à une pose spécifique (la vue de face) et contient 70 images.

FASSEG ne permet cependant pas de distinguer certaines régions du visage (c'est-à-dire l'œil gauche et l'œil droit, le sourcil gauche et le sourcil droit). Nous avons donc affiné les annotations afin de donner un label unique à ces régions (voir figure 5).

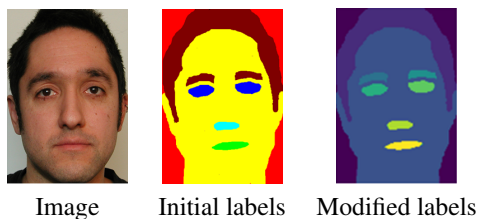


FIGURE 5 – Étiquettes FASSEG modifiées.

3.2 Protocole d'évaluation

Pour les expériences, nous considérons le réseau neuronal U-Net [16] qui s'adapte bien à un ensemble d'entraînement avec un petit nombre d'échantillons. Nous avons aussi divisé notre ensemble de données comme suit : 20 images sont utilisées pour l'ensemble d'entraînement de référence, 10 pour l'ensemble de validation de référence et 40 pour l'ensemble de test de référence. De plus, 100 époques sont utilisées pour la formation du réseau.

Le graphe modèle est construit en calculant les distances moyennes (équation 2) entre les différentes régions annotées de l'ensemble d'entraînement. Le paramètre α est choisi empiriquement, sur la base d'observations sur certaines images, et est fixé à 0,4 pour les expériences, mais sans aucune optimisation.

Comme cela a été étudié expérimentalement, une simple mesure de la distance entre les barycentres (centres de gravité) des régions est apparue inappropriée, en raison de la

¹. L'ensemble de données publiques annotées FASSEG peut être téléchargé à l'adresse suivante : <https://github.com/massimomauro/FASSEG-dataset>.

variabilité de la forme de certaines régions (par exemple les cheveux).

Nous évaluons la différence entre la qualité de la segmentation sémantique à la sortie des réseaux de neurones et celle après la mise en correspondance, c'est-à-dire avec l'intégration de l'information structurelle. Nous considérons l'*Intersection over Union* (IoU ou indice de Jaccard) pour évaluer la qualité de nos résultats par rapport à notre annotation manuelle utilisée comme vérité terrain. Cette mesure d'évaluation est utilisée pour comparer les régions au niveau des pixels et également au niveau des boîtes englobantes. Cette comparaison des boîtes englobantes nous permet de quantifier les erreurs de segmentation correspondant à une région principale correcte mais avec des erreurs liées à une ou plusieurs sous-régions éloignées de la région principale, et comprenant peu de pixels. Ces mesures sont effectuées pour chaque classe, la valeur moyenne globale étant également calculée.

Nous étudions également l'impact de la taille de la base d'apprentissage sur la qualité de notre segmentation sémantique. Des expériences sont réalisées pour différentes tailles, exprimées en pourcentage de la base de référence, à savoir 100 % (totalité des 20 images de l'ensemble d'entraînement et totalité des 10 images pour la validation), 75 % (15 images de l'ensemble d'entraînement et 7 images de l'ensemble de validation), 50 % et 25 % (5 images de l'ensemble d'entraînement et 2 pour la validation). Pour les tailles allant de 75 % à 25 %, les expériences sont effectuées 20 fois avec un tirage aléatoire des images utilisées pour l'entraînement et la validation, et les performances moyennes sont retenues.

3.3 Résultats quantitatifs

Les tables 1 et 2 fournissent des résultats respectivement en termes d'indice IoU pixel par pixel et d'indice IoU en termes de boîtes englobantes. Dans l'ensemble, la segmentation sémantique a été améliorée grâce à l'appariement de graphes, comme on peut le constater en moyenne sur l'ensemble des classes.

L'amélioration apparaît plus significative lorsque l'on utilise l'IoU sur les boîtes englobantes des régions (table 2) par rapport à l'IoU sur les régions elles-mêmes (table 1). Cela est dû au fait que, pour une classe donnée, l'IoU sur les boîtes englobantes est très sensible à la répartition spatiale des zones mal classées, même si ces zones sont de petite taille par rapport à l'ensemble de la région. La mesure d'IoU sur les régions est beaucoup moins sensible à ces erreurs, en particulier pour ces zones d'erreur de petite taille (d'autant plus pour des classes de grande taille telles que les cheveux ou le visage, contrairement aux sourcils et aux yeux par exemple). Ainsi, on peut noter, en considérant les boîtes englobantes, une amélioration très significative de 24,4 % pour la classe *sourcil droit* (R-Br) avec une taille de jeu d'apprentissage de 50 % (table 2). L'utilisation de relations spatiales permet d'éviter les régions mal classées et spatialement incohérentes (telles que des cheveux trouvés entre le nez et l'œil droit).

Les deux tables 1 et 2 illustrent également l'influence de la

TABLE 1 – Comparaison, mesurée à l’aide de l’IoU sur les régions, entre notre segmentation manuelle (de référence) et celle fournie par le U-Net uniquement (notée U-Net) ainsi que celle résultant de notre approche (notée U-Net+GM, où GM signifie *graph matching*). Les résultats sont fournis sous forme de moyenne et pour chaque classe : Bg (fond), Hr (cheveux), Fc (visage), L-br (sourcil gauche), R-br (sourcil droit), L-eye (oeil gauche), R-eye (oeil droit), Nose (nez) et Mouth (bouche). Les résultats sont également fournis pour les différentes tailles de base d’apprentissage.

| Ensemble d’apprentissage (%) | Approche | Classes | | | | | | | | | |
|------------------------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | | Moyenne | Bg | Hr | Fc | L-br | R-br | L-eye | R-eye | Nose | Mouth |
| 100 | U-Net | 75,3 | 88,0 | 88,2 | 91,9 | 61,8 | 60,5 | 76,9 | 72,8 | 67,3 | 77,2 |
| | U-Net + GM | 75,4 | 88,0 | 88,9 | 91,8 | 62,1 | 60,7 | 77,0 | 72,8 | 67,3 | 77,2 |
| 75 | U-Net | 74,0 | 88,5 | 85,9 | 91,0 | 60,8 | 56,8 | 75,5 | 72,8 | 64,7 | 77,9 |
| | U-Net + GM | 74,3 | 88,3 | 86,8 | 91,1 | 61,7 | 57,5 | 75,6 | 72,8 | 64,7 | 78,0 |
| 50 | U-Net | 72,0 | 86,0 | 84,9 | 90,9 | 54,6 | 54,2 | 73,5 | 72,4 | 65,9 | 75,4 |
| | U-Net + GM | 73,7 | 86,7 | 86,9 | 91,0 | 59,9 | 57,9 | 74,5 | 72,5 | 66,1 | 75,6 |
| 25 | U-Net | 38,2 | 84,5 | 83,6 | 56,8 | 2,4 | 28,1 | 61,7 | 25,6 | 57,7 | 70,0 |
| | U-Net + GM | 39,8 | 83,8 | 86,4 | 61,0 | 2,7 | 30,6 | 65,7 | 26,5 | 55,1 | 71,4 |

TABLE 2 – Comparaison, mesurée à l’aide de l’indice IoU sur les boîtes englobantes des régions, entre la segmentation manuelle et celle fournie par le U-Net (U-Net) ainsi que celle résultant de notre approche (U-Net+GM, où GM signifie *graph matching*). Les résultats sont fournis sous forme de moyenne et pour chaque classe : Bg (fond), Hr (cheveux), Fc (visage), L-br (sourcil gauche), R-br (sourcil droit), L-eye (oeil gauche), R-eye (oeil droit), Nose (nez) et Mouth (bouche). Les résultats sont également fournis pour les différentes tailles de base d’apprentissage.

| Ensemble d’apprentissage (%) | Approche | Classes | | | | | | | | | |
|------------------------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | | Moyenne | Bg | Hr | Fc | L-br | R-br | L-eye | R-eye | Nose | Mouth |
| 100 | U-Net | 76,0 | 84,0 | 82,5 | 96,1 | 66,3 | 63,6 | 74,0 | 75,1 | 69,0 | 78,2 |
| | U-Net + GM | 78,4 | 84,0 | 92,4 | 96,1 | 66,5 | 68,3 | 79,4 | 75,2 | 70,4 | 78,9 |
| 75 | U-Net | 74,7 | 84,0 | 74,5 | 95,8 | 66,7 | 59,5 | 78,0 | 75,1 | 64,9 | 79,8 |
| | U-Net + GM | 77,5 | 84,0 | 84,9 | 96,1 | 66,8 | 69,2 | 78,8 | 75,0 | 67,5 | 79,9 |
| 50 | U-Net | 68,0 | 85,4 | 75,7 | 94,5 | 52,4 | 41,8 | 77,7 | 66,6 | 63,8 | 70,6 |
| | U-Net + GM | 76,5 | 85,4 | 86,0 | 94,4 | 65,3 | 66,2 | 78,5 | 74,9 | 68,5 | 74,7 |
| 25 | U-Net | 36,3 | 81,2 | 80,1 | 90,5 | 1,7 | 33,7 | 68,4 | 12,8 | 55,1 | 66,9 |
| | U-Net + GM | 42,3 | 78,0 | 90,1 | 82,8 | 2,9 | 34,8 | 65,9 | 27,4 | 59,3 | 70,1 |

réduction de la base d’apprentissage. Il apparaît que plus la base est réduite, plus l’apport des relations spatiales de haut niveau est important (cela étant très significatif dans la table 2).

Il est à noter que, pour une base d’apprentissage de seulement 25 %, en fonction d’un tirage aléatoire réalisé, certains réseaux semblaient incapables de proposer des régions candidates pour certaines classes sur certaines images de test. Cela est dû à la faible représentativité de la base d’apprentissage. Dans ces cas, la mise en correspondance des graphes échoue en raison de l’absence de candidats pour certaines classes, ce qui n’est pas encore géré par notre approche. C’est pourquoi, pour un taux de 25 %, les résultats présentés dans les tables 1 et 2 ignorent ces cas, et les performances rapportées ne sont moyennées que sur des images de test segmentées fournissant au moins une région par classe (48,9 % des images ne sont pas prises en compte).

3.4 Résultats qualitatifs de la segmentation

Les figures 6 et 7 donnent quelques exemples de segmentations sémantiques avec le U-Net et notre approche.

La figure 6 donne quelques exemples représentatifs. Comme on peut l’observer visuellement, les améliorations sont significatives dans de nombreux cas (cf. figure 6 - image 75 % : cheveux en bas du visage). Même pour un petit ensemble de données d’entraînement, les informations structurelles peuvent améliorer la segmentation, parfois seulement partiellement dans le cas de sorties U-Net

vraiment dégradées (cf. figure 6 - image 25 %).

La figure 7 se concentre sur la réduction de la base de données d’apprentissage pour illustrer la robustesse de notre approche. Ces exemples représentatifs illustrent comment la réduction de la base de données détériore la segmentation en utilisant uniquement des réseaux neuronaux profonds. Cela montre comment notre approche, intégrant des contraintes spatiales globales, peut compenser de manière significative la perte de diversité de l’ensemble des données d’apprentissage.

3.5 Complexité algorithmique

La table 3 indique les durées d’exécution mesurées (moyenne sur les 40 images utilisées pour les tests), pour différentes tailles de la base de données d’apprentissage. Il s’agit d’une estimation brute, pour ce domaine d’application, des durées d’exécution sans détailler le coût de chaque étape, et sans aucune tentative d’optimisation du code, écrit en Python (utilisation de `numpy` pour l’algèbre linéaire). A noter que les programmes ont été exécutés sur un noyau Intel I7, sans GPU. Les durées d’exécution sont plus faibles pour une grande taille de base de données d’apprentissage, car le réseau U-Net produit moins d’artefacts, ce qui réduit le nombre de sommets dans G_r et donc le temps d’appariement.

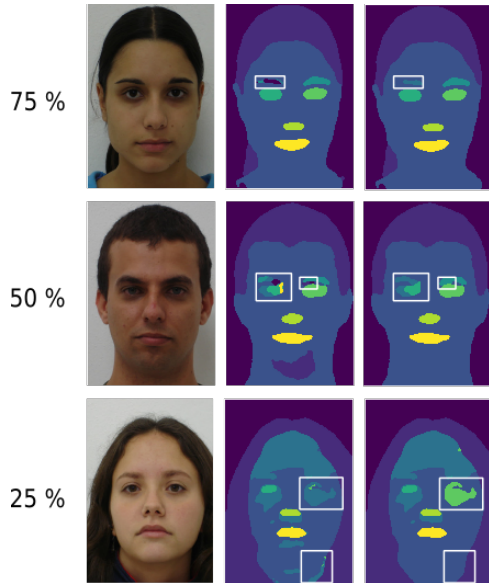


FIGURE 6 – Exemples de résultats de segmentation (image initiale, résultat avec U-Net, résultat avec l’approche proposée), pour différentes tailles de la base d’apprentissage (75%, 50% et 25%).

TABLE 3 – Durée d’exécution de l’ensemble des données de test, en secondes. Moins nous utilisons de données pendant la phase d’apprentissage, plus il faudra de temps pour traiter une image de test, comme le montre l’augmentation du temps moyen.

| Apprentissage (%) | Moyenne | Ecart-type | Médiane | Min | Max |
|-------------------|---------|------------|---------|------|---------|
| 100 | 28,5 | 2,3 | 27,8 | 25,4 | 34,1 |
| 75 | 39,4 | 29,8 | 27,4 | 25,1 | 372,6 |
| 50 | 81,0 | 791,0 | 31,0 | 24,7 | 20165,9 |
| 25 | 171,5 | 599,2 | 49,1 | 25,3 | 7363,8 |

4 Discussion et perspectives

L’approche proposée semble efficace pour améliorer la segmentation sémantique réalisée par le réseau neuronal profond U-Net. L’amélioration semble particulièrement significative pour certaines classes (jusqu’à 24,4 % pour la classe *sourcil droit*, comparé à l’amélioration moyenne de 8,5 % en considérant l’IoU sur les boîtes englobantes avec un pourcentage de la base d’apprentissage de 50 %). Cela est dû à la nature des informations considérées, qui semblent très complémentaires aux informations de bas niveau (au niveau des pixels) considérées par le U-Net, qui ignore intrinsèquement les relations spatiales observées à haut niveau. Un autre point fort de notre approche est sa capacité à compenser les erreurs de segmentation résultant de la réduction de la base de données d’apprentissage. C’est un aspect important car sa taille est une limitation forte de l’apprentissage profond, comme cela a été souligné dans l’introduction.

Bien que prometteuse, notre approche présente quelques limites. Premièrement, notre approche est invariante en

translation et en rotation, mais pas encore en échelle. On pourrait y remédier en introduisant un facteur d’échelle (à estimer automatiquement) dans le processus d’appariement (notamment en ce qui concerne le calcul de la matrice K figurant dans l’équation 3).

Deuxièmement, les occultations partielles dans les images qui pourraient affecter le calcul du graphe G_r , sont des cas qui ne sont pas encore pris en compte dans notre formulation actuelle. En effet, on suppose que chaque sommet (région) correspond nécessairement à une région du modèle. Le traitement des occultations pourrait être géré en assouplissant les hypothèses relatives à l’appariement aux sommets du modèle.

Troisièmement, nous n’avons pas étudié, dans cet exemple applicatif, la robustesse de notre approche vis-à-vis du changement de pose qui pourrait modifier les relations spatiales. Il est à noter que d’autres domaines d’application de notre technique, comme la segmentation d’images médicales en 3D, peuvent ne pas présenter les mêmes défis en pratique que la segmentation de visage présentée ici. Pour supporter une forte variation dans les relations spatiales, une solution pourrait être de considérer un ensemble de graphes modèles représentatifs au lieu d’un seul, avec la difficulté sous-jacente de choisir le modèle approprié lors de la segmentation d’une image.

Enfin, la réduction du temps de calcul peut être cruciale dans certains scénarios d’application. En particulier, la grande complexité de la première étape (formulée comme un QAP) peut impliquer une augmentation considérable du temps de calcul, si le nombre de classes et de régions augmente. Néanmoins, les estimations brutes fournies dans le document sont encourageantes et des améliorations supplémentaires pourraient être recherchées en utilisant du matériel spécialisé (par exemple, GPU).

5 Conclusion

Dans cet article, nous avons proposé une méthode originale combinant les relations spatiales avec un apprentissage profond pour la segmentation sémantique. Les résultats préliminaires montrent le potentiel de cette approche, avec des améliorations significatives dans certains cas tout en nous permettant de réduire la taille de la base d’apprentissage requise.

L’amélioration obtenue est de 8,5% en moyenne, par rapport à la segmentation sémantique résultant du réseau neuronal U-Net, entraîné avec seulement 50% de l’ensemble de données d’apprentissage disponible.

Les travaux futurs se concentreront sur l’invariance par changement d’échelle, une évaluation plus fine de la complexité et sur des expériences dans d’autres domaines d’application.

Remerciements

Ces travaux ont été menés dans le cadre du programme région Atlantic2020 (Recherche, Formation et Innovation en Pays de la Loire), soutenu par la région des Pays de la Loire et le fonds européen de développement.

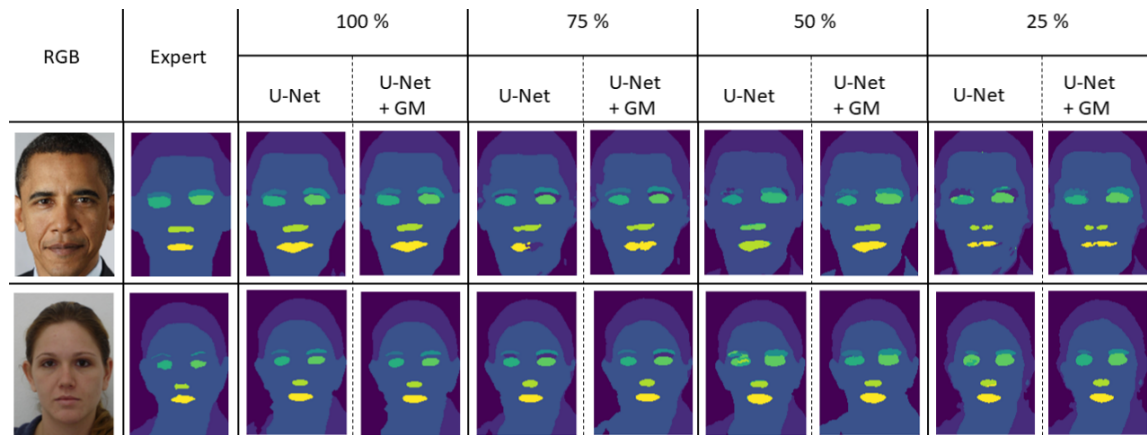


FIGURE 7 – Influence de la taille de la base de données d’apprentissage (de 100 % à 25 %) avec comme exemples des résultats obtenus sur 2 images représentatives.

Références

- [1] I. Bloch. Fuzzy sets for image processing and understanding. *Fuzzy Sets and Systems*, 281 :280–291, 2015.
- [2] O. Colliot, O. Camara, and I. Bloch. Integration of fuzzy spatial relations in deformable models - application to brain MRI segmentation. *Pattern Recognition*, 39 :1401–1414, 2006.
- [3] J.-B. Fasquel, V. Agnus, J. Moreau, L. Soler, and J. Marescaux. An interactive medical image segmentation system based on the optimal management of regions of interest using topological medical knowledge. *Computer Methods and Programs in Biomedicine*, 82 :216–230, 2006.
- [4] J.-B. Fasquel and N. Delanoue. An approach for sequential image interpretation using a priori binary perceptual topological and photometric knowledge and k-means based segmentation. *Journal of the Optical Society of America A*, 35(6) :936–945, 2018.
- [5] J.-B. Fasquel and N. Delanoue. A graph based image interpretation method using a priori qualitative inclusion and photometric relationships. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(5) :1043–1055, 2019.
- [6] H. Gao and S. Ji. Graph U-Nets. In K. Chaudhuri and R. Salakhutdinov, editors, *36th International Conference on Machine Learning*, volume 97, pages 2083–2092, 2019.
- [7] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez. A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing*, 70 :41 – 65, 2018.
- [8] I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.
- [9] P. Goyal and E. Ferrara. Graph embedding techniques, applications, and performance : A survey. *Knowledge-Based Systems*, 151 :78 – 94, 2018.
- [10] K. Khan, M. Mauro, and R. Leonardi. Multi-class semantic segmentation of faces. In *IEEE International Conference on Image Processing*, pages 827–831, 2015.
- [11] K. Khan, M. Mauro, P. Migliorati, and R. Leonardi. Head pose estimation through multi-class face segmentation. In *IEEE International Conference on Multimedia and Expo*, pages 175–180, 2017.
- [12] O. Lezoray and L. Grady. *Image Processing and Analysis with Graphs : Theory and Practice*. Digital Imaging and Computer Vision. CRC Press, 2012.
- [13] Z. Li, L. Zhang, and G. Song. GCN-LASE : Towards adequately incorporating link attributes in graph convolutional networks. In *Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2959–2965, 2019.
- [14] A. Moreno, C.M. Takemura, O. Colliot, O. Camara, and I. Bloch. Using anatomical knowledge expressed as fuzzy constraints to segment the heart in CT images. *Pattern Recognition*, 41(8) :2525 – 2540, 2008.
- [15] O. Nempont, J. Atif, and I. Bloch. A constraint propagation approach to structural model based image segmentation and recognition. *Information Sciences*, 246 :1–27, 2013.
- [16] O. Ronneberger, P. Fischer, and T. Brox. U-Net : Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.
- [17] F. Zhou and F. De la Torre. Factorized graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9) :1774–1789, 2016.

Modélisation sur graphe de la singularité de trajectoires contraintes

Hippolyte Dubois¹, Patrick Le Callet¹, Antoine Coutrot¹

¹ LS2N, Université de Nantes, CNRS, F-44000 Nantes

2-3 Juillet 2020

Résumé

Le développement du traitement du signal sur graphe ces dernières années a ouvert de nouvelles perspectives pour de nombreux domaines de recherche. Cependant, lorsque les données ne sont pas explicitement définies sur un graphe, il peut être compliqué de définir un graphe permettant de représenter au mieux la structure des dites données. Dans cet article, nous nous concentrons sur le problème particulier de la conception de graphe pour l'analyse de trajectoires spatiales. Nous présentons une méthode ad-hoc issue de l'état de l'art, ainsi qu'une méthode nouvelle conçue pour les trajectoires dans des espaces non linéaires contraints, se basant sur une extraction automatique de zones d'intérêt.

Mots-clés

trajectoires, zones d'intérêt, scanpaths, traitement du signal, signal sur graphe

Abstract

The recent developments of graph signal processing opened new possibilities for many research fields. However, in some cases where the data is not originally defined on a graph, it can be challenging to define a graph that models the data domain the best. In this article, we focus on the issue of graph generation for trajectory analysis. We present an ad-hoc method from the literature, as well as a novel approach designed for trajectories defined on non-linear constrained domains, based on the automatic extraction of areas of interest.

Keywords

trajectories, areas of interest, scanpaths, signal processing, graph signal processing

1 Introduction

Pour modéliser des trajectoires spatiales, c'est à dire des séries temporelles définies dans un espace en deux dimension, comme par exemple les trajectoires de mouvements oculaires sur une image, il peut être utile d'extraire des zones d'intérêt du domaine.

Dans cet article, nous présenterons deux approches employées pour ce faire, nous montrerons leurs limitations, puis proposerons une technique améliorée qui y réponde.

2 Extraction de zones d'intérêt

2.1 Annotation sémantique

On retrouve dans la littérature des approches sémantiques à l'identification de zones d'intérêt. Pour analyser des trajectoires de mouvements oculaires sur des visages, [4] utilise un algorithme de reconnaissance d'éléments de visages, les coordonnées de ces éléments servant ensuite à construire les zones d'intérêt. Pour analyser les trajectoires de visiteurs dans un musée, [3] propose de définir à la main, de façon savante, des éléments sémantiquement signifiants de l'espace, en y donnant une structure hiérarchique avec plusieurs niveaux de granularité. Ces approches nécessitent d'avoir une connaissance des éléments sémantiquement importants du domaine à modéliser, ce qui n'est pas toujours possible.

2.2 Annotation automatique

Pour remédier à ce problème, d'autres approches, plus automatiques celles ci, existent dans la littérature. La plupart reposent sur l'extraction de points souvent visités par les trajectoires. Par exemple, [2] propose un modèle qui permet de prendre en compte les dimensions spatiales et temporelles des données, en projetant les trajectoires dans un cube x, y, t et en extrayant de chaque polyèdre à l'intérieur de ce cube un maximum local puis en joignant ceux trop proches.

2.3 Limitations

Jusqu'à présent, la plupart des approches proposées dans la littérature modélisent des trajectoires effectuées dans un espace euclidien sans obstacle. C'est par exemple le cas des trajectoires oculaires, ou de celles d'animaux se déplaçant en plein air. Ici nous nous attaquons au problème de trajectoires effectuées dans un espace contraints, c-a-d avec des obstacles. Ces contraintes conduisent à des trajectoires plus stéréotypées, rendant les maxima locaux de densité non informatif car visités par chacun. Pour dépasser ce problème, nous proposons de prendre en compte les comportements marginaux, contenant la singularité du signal.

3 Modèle proposé

Il convient donc de construire nos modèles en prenant en compte les comportements marginaux, les singularités du signal.

Pour donner des exemples de résultats obtenus via cette

technique, nous l’appliquons sur des données extraites du jeu de données Sea Hero Quest [1], en particulier de 15548 trajectoires prises sur le niveau 48.

3.1 Définition des noeuds

Pour ce faire, deux types de noeuds sont inclus au graphe. Les noeuds *normaux* qui représentent le signal moyen, le comportement le plus commun, et les noeuds *marginiaux* qui eux représentent le signal singulier, le comportement marginal.

$$\mathcal{E} = \mathcal{E}_{norm} \oplus \mathcal{E}_{marg} \quad (1)$$

$$\mathcal{E}_{norm} = LocalMaxPeaks(D) \quad (2)$$

$$\mathcal{E}_{marg} = LocalMaxPeaks(1 - (D > m)) \quad (3)$$

$$D_{xyt} = \frac{1}{N} \sum_i \exp\left(\frac{-\|p_i - \{x, y, t\}\|^2}{2\sigma^2}\right) \quad (4)$$

Avec p un point, σ le paramètre de proximité (plus il est bas, moins les points éloignés sont tolérés), et m le seuil ou delà duquel calculer les minimums.

Les noeuds normaux sont les maximums locaux du cube de chaleur D dont chaque valeur D_{xyt} est la proximité moyenne de tous les points de toutes les trajectoire au point de coordonnée $\{x, y\}$ à l’instant t .

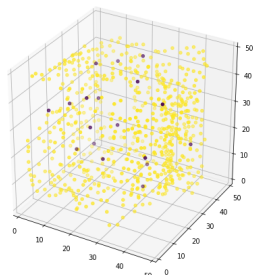


FIGURE 1 – Centres des zones d’intérêt extraites. En jaune, les points marginaux, en violet, les points normaux.

Une fois ces noeuds définis, la définition des arrêtes du graphe est importante.

3.2 Définition des arrêtes

Arrêtes de passage On peut définir la condition à l’existence d’une arrête entre deux noeuds à l’existence d’une trajectoire passant de la zone du premier noeud à la zone du second. Cette approche permet de modéliser les relations entre les différentes zones d’intérêt du point de vue du comportement des agents qui ont généré les trajectoires.

Structure arborescente On peut définir un graphe hiérarchique sur trois niveaux, avec en feuilles les noeuds marginaux, reliés au noeud normal le plus proche, avec un noeud racine auquel sont reliés les noeuds normaux. Cette structure est complétée par un second jeu d’arrête, qui relie entre elles les feuilles d’un même noeud. Cette structure présente l’avantage de permettre de simplifier l’implémentation d’un algorithme de convolution sur graphe.

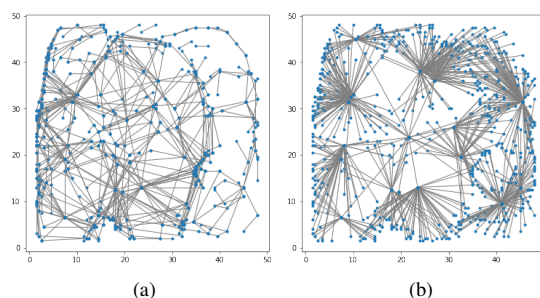


FIGURE 2 – Projections dans xy des graphes ainsi générés. (a) Graphe de passage. Seules les 15% des arrêtes les plus visitées sont affichées pour rendre le graphe plus lisible. (b) Graphe à structure arborescente. Les arrêtes inter-feuilles sont cachées puisque trop nombreuses.

3.3 Apports

En permettant une meilleure couverture du domaine dans lequel les trajectoires sont définies, cette approche devrait permettre de mieux représenter la singularité des données grâce aux noeuds marginaux, et la structure graphe de modéliser les dépendances spatio-temporelles propres aux trajectoires. Cette thèse devra être validée par de futures expérimentations.

Remerciements

Ce projet a été partiellement financé par le *RFI Atlantic 2020* et le *RFI Ouest Industrie Creative*, programmes de la région Pays de la Loire.

Références

- [1] Antoine Coutrot, Ricardo Silva, Ed Manley, Will de Cothi, Saber Sami, Véronique D. Bohbot, Jan M. Wiener, Christoph Hölscher, Ruth C. Dalton, Michael Hornberger, and Hugo J. Spiers. Global Determinants of Navigation Ability. *Current Biology*, 28(17) :2861–2866.e4, September 2018.
- [2] Gautier Drusch, J. M. Christian Bastien, and Stéphane Paris. Analysing eye-tracking data : From scanpaths and heatmaps to the dynamic visualisation of areas of interest. In *International Conference on Applied Human Factors and Ergonomics*, Krakow, Poland, 2014.
- [3] Alexandros Kontarinis, Karine Zeitouni, Claudia Marinica, Dan Vodislav, and Dimitris Kotzinos. Towards a Semantic Indoor Trajectory Model. In *2nd International Workshop on "Big Mobility Data Analytics" (BMDA) with EDBT 2019*, Lisbon, Portugal, March 2019.
- [4] Michał Król and Magdalena Ewa Król. A Novel Eye Movement Data Transformation Technique that Preserves Temporal Information : A Demonstration in a Face Processing Task. *Sensors (Basel)*, 19(10), May 2019.

Un modèle de jeux de coalitions stochastiques répétés

Josselin Guéron¹ et Grégory Bonnet¹

¹ Normandie Université, UNICAEN, ENSICAEN, CNRS UMR 6072, CREYC, 14000 Caen, France

{prenom.nom}@unicaen.fr

Résumé

Les jeux de coalitions sont des modèles de coopération où des agents égoïstes doivent former des groupes leur permettant de maximiser leur utilité. Dans ces modèles, il est fait l'hypothèse que l'utilité d'une coalition est fixe et connue des agents. Dans cet article, nous proposons de lever cette hypothèse et nous proposons un modèle de jeu de coalitions dont la fonction caractéristique est stochastique. Ainsi, les agents ne la connaissent pas a priori et doivent donc l'apprendre. Pour cela, ce jeu de coalitions devient un jeu répété où les agents estiment au fur et à mesure de leurs interactions la fonction caractéristique. Cependant comme les coalitions formées doivent satisfaire une propriété de stabilité (car les agents sont égoïstes), nous proposons un nouveau concept de solution permettant d'intégrer à l'utilité estimée d'une coalition un biais d'exploration. Nous expérimentons deux types de biais : un biais individuel et un biais collectif.

Mots-clés

Décision séquentielle, Formation de coalitions, Théorie des jeux coopératifs, Systèmes multi-agents.

Abstract

Coalition games are models of cooperation where selfish agents must form groups (coalitions) to maximize their utility. In these models, it is assumed that the utility of a coalition is fixed and known. In this article, we relax those hypotheses and we propose a model of coalition formation whose characteristic function is stochastic. The agents do not know it a priori and must learn it through a repeated game : the agents estimate the characteristic function with their interactions. However as the decision to form coalitions must satisfy a stability property (because agents are selfish), we propose a new solution concept which integrates the estimated utility of a coalition and a bias of exploration. We experiment two kinds of bias : one individual bias and one collective bias.

Keywords

Sequential decision, Coalition formation, Cooperative game theory, Multi-agent systems.

1 Introduction

Dans les systèmes multi-agents, les agents individuels n'ont pas nécessairement la capacité de réaliser certaines

tâches seuls. De ce fait, chaque agent peut alors se poser la question de la coopération. Dans ce cadre, une approche de la théorie des jeux coopératifs tente de répondre à cette question, grâce à la formation de coalitions. Cependant, une grande majorité des travaux utilisant la formation de coalitions font deux hypothèses fortes. La première est que les agents possèdent une connaissance parfaite a priori de l'intérêt à former une coalition, c'est-à-dire du gain qu'il y a à la former. La deuxième hypothèse est que ce gain est déterministe. Ces deux hypothèses éloignent le modèle d'une situation réelle car il semble plus réaliste de connaître le gain exact apporté par une certaine coopération seulement a posteriori, et qu'en cas de répétition de cette coopération, le gain ne soit pas toujours le même, comme par exemple si les agents sont plus ou moins efficaces dans leurs tâches. Par exemple, nous pouvons considérer des entités morales devant former de manière répétée des consortiums afin de travailler temporairement sur des projets : ces formations de consortium se répètent sans pour autant que les acteurs ne changent. Toutefois, la qualité des résultats produits par chaque consortium va de facteurs internes comme la compétence individuelle des entités et leur capacité à mieux interagir avec certaines mais pas d'autres, et d'externalités indépendantes des consortiums formés. Il semble donc approprié de s'affranchir de ces deux hypothèses mais le faire soulève des questions. Par exemple, si les agents n'ont plus de connaissances a priori sur les coalitions, comment les obtenir ? Une piste prometteuse pour modéliser un jeu de coalitions où ces hypothèses sont relâchées est de s'orienter vers les jeux répétés, qui permettent d'observer le même jeu – ici un jeu de coalitions donc – de façon répétée, ce qui permet aux agents d'observer l'état du jeu à différents instants, et donc de pouvoir rassembler des informations sur le jeu. Par exemple, dans un jeu de coalitions où les coalitions ne produisent pas un gain déterministe, il serait possible pour les agents de faire une estimation de ce gain, et ce de manière de plus en plus précise avec le temps, ce qui leur permettrait de décider d'une manière plus certaine quelles coalitions former.

L'utilisation des jeux répétés permettrait donc d'observer et ainsi acquérir de la connaissance sur les gains des coalitions, même si ce gain est stochastique. Néanmoins, tout ceci impacte également le but premier de la formation de coalitions, c'est-à-dire décider du partitionnement des agents en coalitions. Dans un cadre où l'on doit apprendre l'ensemble des utilités des coalitions, où celles-ci sont sto-

chastiques, comment décider des coalitions à former ?

Dans un cadre classique de formation de coalitions, la décision dépend de la notion de concept de solution. Lorsqu'une solution est proposée, celle-ci doit être acceptée par tous les agents selon un critère d'optimalité et de stabilité. Dès lors où nous proposons de relâcher les deux hypothèses précédemment évoquées, se posent de nouvelles questions. Comment les agents peuvent-ils privilégier la formation d'une certaine coalition plutôt qu'une autre en fonction de ce qu'ils savent sur ces coalitions ? À partir de quel moment considèrent-ils connaître suffisamment une coalition pour en estimer correctement l'utilité ? Comment décider collectivement d'interactions communes acceptées par tous les agents ?

Nous proposons alors dans cet article un modèle de jeu de coalitions stochastique répété, dont le but est d'apprendre une fonction caractéristique stochastique, et de décider séquentiellement de structures de coalitions stables à former. Dans ce modèle, nous proposons des redéfinitions de concepts de solutions existants, afin de permettre aux agents de former des coalitions stables tout en acceptant d'en former pour acquérir de la connaissance sur les utilités probables que produiraient les coalitions. Cela consiste donc à introduire dans le jeu de coalitions répété une nouvelle notion de stabilité permettant un compromis entre exploiter un gain plutôt connu ou explorer des coalitions dont le gain est inconnu ou très incertain.

Cet article est structuré comme suit. Dans la section 2, nous présentons les notions fondamentales liées aux jeux de coalitions, puis nous passons en revue des travaux s'intéressant à la répétition de ces jeux et enfin nous introduisons des notions liées à l'apprentissage et à la décision séquentielle. Dans la section 3, nous décrivons un modèle de jeu de coalitions stochastique, permettant la formation de coalitions répétée dans laquelle l'apprentissage et la décision commune sont des points essentiels. La section 4 présentera un exemple illustratif de notre modèle. Enfin, la section 5 sera consacrée à la proposition et la mise en pratique d'un protocole d'expérimentation afin d'évaluer notre modèle, ainsi qu'à l'analyse des résultats.

2 État de l'art

Dans cette section, nous présentons les notions fondamentales sur les jeux de coalitions [14], les jeux de coalitions répétés ainsi que de la stochasticité dans les jeux de coalitions.

2.1 Formation de coalitions

Dans un système multi-agents, il peut arriver que les agents ne puissent pas réaliser certaines tâches seuls, ou bien gagneraient à coopérer avec d'autres. Lorsque ces agents coopèrent, ils forment une *coalition*. Un partitionnement en coalitions disjointes d'un ensemble d'agents est appelé *structure de coalitions*.

Définition 1 (Jeu de coalitions) Un jeu de coalitions est un tuple $\mathcal{G} = \langle N, v \rangle$ où :

- $N = \{a_1, \dots, a_n\}$ est un ensemble d'agents,

- $v : 2^N \rightarrow \mathbb{R}$ est une fonction caractéristique qui à chaque coalition associe une valeur réelle, appelée utilité de la coalition et notée $v(C_k)$ où $C_k \subseteq N$.

Nous considérons dans cet article des jeux de coalitions à utilité transférable, c'est-à-dire où les agents doivent décider de la répartition de l'utilité de coalition entre les membres de cette dernière [9]. La *solution* à un jeu de coalitions à utilité transférable est définie comme suit.

Définition 2 (Solution d'un jeu de coalitions) Une solution à un jeu de coalitions \mathcal{G} est un tuple $S_{\mathcal{G}} = \langle C, \vec{x} \rangle$ où :

- C est une structure de coalitions de N ,
- $\vec{x} = \{x_1, \dots, x_n\}$ est un vecteur de gain des agents telle que $x_1 \geq 0$ est le gain de l'agent a_i .

Les agents étant égoïstes, ils souhaitent obtenir un gain maximum. Lorsqu'une solution est proposée, tous les agents doivent l'accepter, c'est-à-dire qu'ils ne doivent pas souhaiter former ou rejoindre une autre coalition où ils pourraient gagner davantage. C'est pourquoi la solution d'un jeu de coalitions doit être stable, ce qu'elle est lorsqu'elle appartient à un *concept de solution*. Un concept de solution est l'ensemble des solutions à un jeu de coalitions qui respectent un certain concept de stabilité prédéfini. De nombreux concepts ont été proposés dans la littérature, comme le nucléole ou le noyau [5], mais nous ne sommes intéressés dans cet article qu'au concept de *cœur* et sa généralisation l' ϵ -*cœur* [13, 17].

Le cœur est l'ensemble des solutions $\langle C, \vec{x} \rangle$ pour lesquelles aucune autre coalition qui pourrait être formée ne permet de produire une somme de gains supérieure à celle que les agents obtiennent avec \vec{x} . Si le cœur peut être vide, il existe la variante non vide, appelée ϵ -cœur. Cette variante permet aux agents de faire une concession, c'est-à-dire accepter de réduire leur gain de ϵ , afin de former une structure de coalitions stable.

Définition 3 (ϵ -cœur) Une solution $\langle C, \vec{x} \rangle$ appartient au ϵ -cœur si, et seulement si :

$$\forall C \subseteq N, x(C) \geq v(C) - \epsilon \text{ avec } x(C) = \sum_{i \in C} x_i$$

Le nucléole est un concept de solution comparant lexicographiquement des vecteurs de déficit ordonnés, représentant le désir que les agents pourraient avoir à rejoindre des coalitions autres que la grande coalition. Il ne peut pas être vide et est composé d'un unique élément. Le noyau est quant à lui un concept de solution calculant l'influence qu'un agent peut avoir sur un autre afin qu'il lui cède une part de son gain, sans quoi le premier pourrait aller former une autre coalition, laissant le deuxième avec un gain moindre. Or en première instance, notre problématique est d'apprendre la fonction caractéristique avant de se poser la question de la répartition. Le noyau risque donc d'ajouter de la complexité au modèle sans que cela ne soit utile et, comme l'unicité du nucléole peut-être vu comme un frein à l'apprentissage, nous considérons plutôt l' ϵ -cœur.

2.2 Fonctions caractéristiques stochastiques

Comme évoqué précédemment, le fait que l'utilité d'une coalition soit déterministe n'est pas toujours réaliste. Dans la littérature, quelques travaux ont alors proposé de relâcher cette contrainte et d'intégrer de la stochasticité dans les fonctions caractéristiques. Toutefois, ce que représente cet aléa diffère selon les modèles.

Ieong et Shoham ont intégré une distribution de probabilité sur des mondes représentant des jeux de coalitions ayant chacun une fonction caractéristique déterministe [11]. Le vecteur de gain devient un ensemble de contrats, représentant la répartition de l'utilité des coalitions dans chaque monde possible, sur lequel les agents expriment une relation de préférence. Les auteurs redéfinissent des concepts de solution sur la base de ces préférences et de la temporalité des observations des agents : concept *ex-ante* (avant l'observation du monde véritable), concept *ex-interim* (après réduction des mondes possibles mais avant l'observation du monde véritable), concept *ex-post* (après l'observation du monde véritable).

Chalkiadakis et Boutilier considèrent une fonction caractéristique déterministe modélisée dans un environnement stochastique avec des processus de décision markoviens partiellement observables [4]. Les agents ont des croyances sur les capacités des autres agents et une même structure de coalition peut amener à différents états du monde. Dans ce modèle, les auteurs considèrent un concept de stabilité appelé cœur bayésien [3, 6] – qui est un cœur fondé sur les vecteurs d'espérance de gain dépendants des coalitions formées et des actions effectuées par celles-ci – et ils utilisent une variante de l'équation de Bellman afin de maximiser l'espérance de la somme des gains des agents.

Charnes et Granot ont simplement considéré que la valeur d'une coalition est une variable aléatoire [8]. La fonction caractéristique est alors réécrite en $v : 2^N \rightarrow \mathcal{X}_{2^N}$. Dans ce modèle, ils calculent leurs vecteurs de gains en associant à chaque agent le fractile de la loi normale estimée définissant la variable aléatoire de la coalition dans laquelle il se trouve, par rapport au nombre d'agent dans cette même coalition. Ils utilisent une variante du concept du nucléole, le *nucléole-a priori*, afin de l'utiliser dans ce contexte stochastique [7].

Dans cet article, nous souhaitons que notre modèle puisse représenter des situations de stochasticité hétérogène, par exemple, où l'incertitude pourrait venir de la fiabilité des agents qui la compose pour une coalition, et être due à l'environnement pour une autre. Les deux premiers modèles présentés ci-dessus ne répondent pas à un tel souhait, la stochasticité venant d'éléments particuliers précis. Le modèle de Ieong et Shoham nécessite de passer par la définition de mondes dans lesquels les fonctions caractéristiques sont déterministes, et la nature de la stochasticité ne peut venir que de l'incertitude sur le monde réel. Si cela permet de modéliser des fonctions caractéristiques stochastiques, cela ne correspond pas à une nature hétérogène de l'aléa. Le modèle de Chalkiadakis et Boutilier fait, lui, reposer l'incertitude sur l'environnement et les croyances sur les

compétences des agents. Ce modèle est plus permissif que le précédent mais pas encore assez général de notre point de vue car les compétences des agents sont toujours déterministes en soi. C'est pourquoi nous nous positionnons dans la continuité des travaux de Charnes et Granot, ceux-ci permettant par l'utilisation d'une simple variable aléatoire de modéliser l'hétérogénéité de la stochasticité. Toutefois, comme indiqué précédemment et contrairement à Charnes et Granot, nous nous intéressons au concept de l' ϵ -cœur et non pas du nucléole.

2.3 Jeux de coalitions répétés

Si nous nous affranchissons également de l'hypothèse de connaissance parfaite de la fonction caractéristique qu'elle soit stochastique ou non, les agents ne savent donc rien a priori des utilités produites par les coalitions. Il devient alors intéressant de passer dans un cadre répété. La théorie des jeux répétés [1] nous permet de jouer un jeu à plusieurs reprises dans le temps avec les mêmes paramètres (agents, fonction caractéristique, etc.) afin d'observer les utilités générées par les coalitions, d'apprendre une estimation de la fonction caractéristique et de trouver une solution optimale stable dans le temps.

Par exemple Blankenburg *et al.* propose un modèle de jeu de coalitions répété ainsi qu'un protocole afin de déterminer quels agents sont dignes de confiance – c'est-à-dire fiables dans l'exécution de leur travail – pour former une coalition [2]. Ils proposent un protocole en quatre étapes : (1) communication des croyances sur les agents et des utilités observées des coalitions ; (2) formation des coalitions selon une variante du noyau ; (3) paiement des agents avant la réalisation de la tâche pour les inciter à adopter une bonne conduite ; (4) exécution de tâches et évaluation. Répéter le jeu consiste à exécuter ce protocole, formant diverses coalitions et en évaluant le résultat a posteriori. Ainsi, les agents apprennent la fiabilité des autres en observant les utilités des coalitions auxquelles ils participent. Nous pouvons également citer à nouveau Chalkiadakis et Boutilier [4] dont le modèle repose également sur un processus d'apprentissage par renforcement bayésien, qui peut être vu comme un jeu répété. À chaque pas de temps, les agents doivent choisir de former une coalition. Une fois les coalitions formées, ces dernières réalisent une action qui provoque une transition stochastique vers un état du monde. Les agents observent ce qui s'est produit et mettent à jour leur état de croyance. Cela leur permet d'apprendre d'une part les compétences des autres agents mais également le modèle de transition stochastique entre les états.

Ces deux modèles sont intéressants mais s'intéressent essentiellement à l'apprentissage de compétences ou des fiabilités individuelles influant sur une fonction caractéristique déterministe. Aussi, nous allons plutôt nous éloigner de ces modèles pour apprendre une fonction caractéristique stochastique comme définie par Charnes et Garnot. Dans les sections suivantes, nous présentons notre modèle, puis un protocole expérimental permettant de le mettre en pratique, et enfin de premiers résultats.

3 Un modèle pour les jeux de coalitions répétés stochastiques

Nous définissons un jeu répété de coalitions à utilité transférable avec fonction caractéristique stochastique.

Définition 4 (Jeu de coalitions stochastique répété)

Soit un jeu de coalitions stochastique répété (RSCG) $\mathcal{G} = \langle N, \mathbb{T}, v \rangle$ où :

- $N = \{a_1 \dots a_n\}$ est un ensemble d'agents,
- $\mathbb{T} \subset \mathbb{N}^+$ est un ensemble de pas de temps distincts,
- $v : 2^N \rightarrow \mathbb{R}^2$ est une fonction caractéristique qui à toute coalition associe deux réels. Pour une coalition donnée $C \subseteq 2^N$, nous notons $v(C) = (\mu_C, \sigma_C)$. Cette fonction caractéristique est inconnue des agents.

À chaque pas de temps, les agents de N doivent décider d'une solution au jeu, sachant qu'ils ne connaissent pas a priori la fonction caractéristique v . Une solution est, comme dans un cadre déterministe, un couple d'une structure de coalitions et un vecteur de gain. Toutefois, le gain des agents est un gain estimé *ex ante* à partir de ce qu'ils connaissent de v .

Définition 5 (Solution d'un RSCG) Une solution S^t au pas de temps $t \in \mathbb{T}$ à un RSCG \mathcal{G} est un couple $S^t = (C^t, \bar{x}^t)$ tel que :

- C^t est une structure de coalitions (partition disjointe) de N ,
- $\bar{x}^t = \{x_1^t, \dots, x_n^t\}$ est un vecteur de gain tel que $x_i^t \geq 0$ est le gain de l'agent a_i calculé en fonction de l'utilité estimée de la coalition dont il fait partie dans la structure C^t .

Comme dit précédemment, le gain des agents pour une solution donnée est une estimation. Une fois une solution trouvée et les coalitions formées, les utilités réelles produites par les coalitions sont le résultat de processus stochastiques paramétrés par la fonction caractéristique.

Définition 6 (Utilité d'une coalition) L'utilité $g(C, t)$ d'une coalition $C \subseteq 2^N$ formée au pas de temps $t \in \mathbb{T}$ est une variable aléatoire suivant une loi normale $\mathcal{N}(\mu, \sigma^2)$ telle que $\mu = \mu_C$ et $\sigma = \sigma_C$.

Lorsque l'utilité d'une coalition est produite, elle est observée par les agents.

Définition 7 (Observations) Soit $\mathcal{O}_t = \{(C, t', g(C, t')) : C \subseteq 2^N, t' \in \mathbb{T}, t' < t\}$ un ensemble d'observations au temps t correspondant à l'ensemble des coalitions formées aux pas de temps précédents t ainsi que leur utilité réelle produite. Ainsi, sachant une solution S^t d'un RSCG au temps t , $\mathcal{O}_{t+1} = \mathcal{O}_t \cup \{(C, t, g(C, t)) : \forall C \in C^t\}$.

Dans la suite, notons $\mathcal{O}_t(C)$ l'ensemble des observations au temps t associées à la coalition $C \subseteq 2^N$. Comme l'incertitude se situe sur l'utilité produite par les coalitions une

fois formées, une solution doit prendre en considération cette incertitude pour être stable. Nous proposons d'adapter le concept de solution du ϵ -cœur en considérant que la valeur d'une coalition, c'est-à-dire son intérêt à être formée à un pas de temps donné, dépend de deux éléments : une estimation de son utilité dont découle directement les gains des agents et un biais, à la manière d'un biais UCB dans les bandits manchots [10, 12], représentant un intérêt à former la coalition pour obtenir plus d'informations sur son utilité réelle.

Définition 8 (Estimation de l'utilité) Soit $\mathbb{E}(C, t)$ l'estimation de l'utilité d'une coalition $C \subseteq 2^N$ au temps $t \in \mathbb{T}$. Cette estimation est définie comme l'espérance des utilités de $\mathcal{O}_t(C)$.

Définition 9 (Biais d'exploration) Soit $UCB(C, t)$ un biais décrivant l'intérêt que peuvent avoir des agents à former une coalition $C \subseteq 2^N$ afin d'obtenir une estimation de son utilité plus précise aux pas de temps suivant. Ce biais est défini comme suit :

$$UCB(C, t) = \sqrt{\frac{2 \cdot \log(|\mathcal{O}_t| + 1)}{|\mathcal{O}_t(C)| + 1}}$$

En nous inspirant du concept de solution du cœur (voir Définition 3), former une structure de coalition stable signifie qu'il n'existe pas de coalitions ne faisant pas partie de la structure telle que sa valeur estimée et son biais d'exploration soit plus important que les gains estimés par les agents dans la structure et leur intérêt à explorer cette structure. Or, du point de vue des agents, former une structure de coalition peut leur rapporter un gain d'information, d'un point de vue individuel ou collectif. Un biais individuel représentera le fait que l'exploration d'une coalition rapporte de l'information aux agents personnellement sans que ceci ne prennent en compte les autres agents, tandis que le biais collectif désigne un apport commun, qui profite à tous les agents impliqués par la décision.

Définition 10 (Biais d'exploration collectif) Soit C_i la coalition à laquelle appartient l'agent a_i dans une structure de coalition C . Le biais d'exploration d'une coalition $C \subseteq 2^N$ est un biais collectif lorsqu'il est défini comme suit :

$$UCB^C(C, t) = \sum_{a_i \in C} UCB(C_i, t)$$

Définition 11 (Biais d'exploration individuel) Soit C_i la coalition à laquelle appartient l'agent a_i dans une structure de coalition C . Le biais d'exploration est un biais individuel lorsqu'il est défini comme suit :

$$UCB^C(C, t) = \sum_{a_i \in C} \frac{UCB(C_i, t)}{|C_i|}$$

Ainsi, une solution d'un RSCG est *stable* si, et seulement si, il n'existe pas de coalitions qu'un groupe d'agents pourrait former, dont l'estimation de l'utilité plus le biais d'exploration soit supérieur à la somme des gains individuel

des agents dans cette solution plus le biais d'exploration individuel ou collectif de la structure. Formellement :

Définition 12 (Solution stable) La solution $S^t = (C^t, \bar{x}^t)$ d'un RSCG est stable si, et seulement si :

$$\forall C \in \mathcal{N}, x^t(C) + UCB^C(C, t) \geq \mathbb{E}(C, t) - \epsilon + UCB(C, t)$$

tel que :

$$x^t(C) = \sum_{a_i \in C} x_i^t$$

Une solution est collectivement stable si $UCB^C(C, t)$ est un biais d'exploration collectif. Elle est une solution individuellement stable si $UCB^C(C, t)$ est un biais d'exploration individuel.

4 Expérimentations

Afin de savoir si le concept de solution proposé en Définition 12 est intéressant pour l'apprentissage collectif d'une fonction caractéristique, nous procédons empiriquement et comparons les solutions collectivement stables avec les solutions individuellement stables. Nous générons des jeux aléatoires que des agents vont jouer de manière répétée et observons l'évolution des connaissances des agents et l'optimalité des solutions retenues.

4.1 Protocole expérimental

Nous définissons donc un protocole d'expérimentation afin d'implémenter des jeux de coalitions répétés à fonction caractéristique stochastique, et observons leurs évolutions par des mesures de performances mettant en lumière les connaissances des agents et les conséquences de leurs actions à chaque répétition.

Nous générons des jeux avec 5, 6 et 7 agents (pour respectivement 52, 203 et 877 structures de coalitions possibles). Les fonctions caractéristiques de ces jeux sont générées aléatoirement. L'espérance des valeurs de coalitions sont générés selon le modèle NDCS [15] : les paramètres μ de la fonction caractéristique sont tirés aléatoirement suivant la loi normale $\mathcal{N}(|C|, \sqrt{|C|})$. La variance est, quant à elle, générée de sorte à ce qu'elle croisse avec la taille des coalitions tout en évitant les valeurs trop élevées : les paramètres σ^2 sont donc tirés uniformément dans l'intervalle $[\log(|C|), \sqrt{|C|}]$. Une expérimentation consiste à considérer des solutions collectivement stables ou des solutions individuellement stables. Chaque expérimentation se déroule sur 500 pas de temps, et nous effectuons 40 expérimentations avec des paramètres μ et σ^2 différents. Les résultats présentés en Section 4.3 sont la moyenne des résultats obtenus sur ces 40 expérimentations.

Comme le modèle proposé en Section 3 n'est pas un modèle de processus de formation de coalitions, voici celui que nous utilisons pour trouver une solution stable :

1. Nous fixons ϵ à 0.
2. Nous calculons toutes les solutions approximées possibles, c'est-à-dire que pour toutes les structures de coalitions possibles, nous calculons un nombre fini de vecteurs de gains.

3. Nous parcourons ensuite l'ensemble des solutions proposées afin de ne garder que les solutions stables au sens de notre concept de solution.
4. Si aucune des solutions calculées n'est stable, nous recommençons l'étape 3 en ayant incrémenté ϵ . Notons qu'il existe toujours un ϵ suffisamment grand pour qu'il existe au moins une solution stable [13, 5].
5. Une fois trouvée au moins une solution faisant partie du concept de solution, nous en choisissons une aléatoirement : les coalitions sont formées et produisent une utilité réelle.
6. Les agents forment les coalitions de la solution choisie et observent toutes les utilités produites ainsi que les répartitions réelles des gains. Les agents mettent donc à jour leur *connaissance commune* en estimant à nouveau les paramètres μ et σ^2 de la fonction caractéristique pour chaque coalition qui vient d'être formée.

4.2 Mesures de performance

Afin d'évaluer notre modèle, nous utilisons deux mesures : une première qui mesure empiriquement l'efficacité des décisions prises (vue comme l'optimalité des solutions stables trouvées) à travers le temps, et une deuxième qui mesure l'évolution des connaissances des agents, c'est-à-dire l'erreur d'apprentissage (vue comme la différence entre fonction caractéristique réelle et estimée). Ces deux mesures sont :

- Le *regret cumulé collectif* de la solution stable à chaque pas de temps. Le regret cumulé à l'instant t est la somme des différences entre le bien-être social maximal espéré (somme maximale des utilités espérées réelles des structures de coalitions) et la somme des utilités espérées réelles des coalitions de la structure formée à chaque instant précédent t , ainsi que de ce dernier. Formellement, soit $S^* = (C^*, \bar{x}^*)$ la solution optimale, le regret cumulé collectif à l'instant t , noté R^t , est défini par :

$$R_c^t = \sum_{t'=0}^t \left(\sum_{C^* \in \mathcal{C}^*} \mu_{C^*} - \sum_{C \in \mathcal{C}^{t'}} \mu_C \right)$$

- La *distance entre les fonctions caractéristiques* estimée (\hat{v}) et réelle (v). La distance entre les fonctions caractéristiques est définie par la somme des distances de Hellinger¹ entre les lois normales paramétrées par les couples μ, σ^2 de la fonction caractéristique estimée et les couples μ, σ^2 de la fonction caractéristique réelle, et ce à l'instant t . Formellement, la distance D_H^t entre deux fonctions caractéristiques à l'instant t est définie par :

$$D_H^t = \sum_{C \in \mathcal{C}^{2^N}} H(\mathcal{N}(v(C)), \mathcal{N}(\hat{v}(C)))$$

1. Cette distance statistique quantifie la similarité entre deux distributions de probabilité. Cette distance varie entre 0 et 1 où 0 signifie une identité entre les deux lois.

où $H(\mathcal{N}_1, \mathcal{N}_2)$ est la distance de Hellinger entre deux lois normales.

Afin que ces mesures puissent être commensurables entre les différentes expérimentations n'ayant pas les mêmes paramètres, notamment le nombre d'agents, le regret cumulé collectif est divisé par le nombre d'agents et la distance entre les fonctions caractéristiques est quant à elle divisée par le nombre de coalitions possibles de l'expérimentation.

4.3 Résultats

Les Figures 1.a, 1.b et 1.c montrent respectivement le regret cumulé collectif pour 5, 6 et 7 agents. Les courbes rouges représentent les solutions collectivement stables, tandis que les courbes bleues représentent les solutions individuellement stables.

Nous pouvons observer sur ces figures le rôle important du type de biais d'exploration utilisé suivant le nombre d'agents. En effet, le biais individuel est le seul des deux biais à réduire le regret cumulé lorsqu'augmente le nombre d'agents. Pour 5 agents, le regret cumulé atteint environ 220, tandis qu'il n'est que de 140 pour 6 et 7 agents. Pour le biais collectif en revanche, nous pouvons observer le phénomène inverse : le regret cumulé est d'environ 75 pour 5 agents mais il atteint 110 et 130 environ pour 6 et 7 agents respectivement. Nous avons donc un biais collectif plus performant qu'un biais individuel en ce qui concerne le regret, que ce soit avec 5, 6 ou 7 agents. Toutefois, plus il y a d'agents plus les différences entre les biais se réduisent. Cela peut sans doute être expliqué par le nombre de coalitions augmentant exponentiellement par rapport au nombre d'agents. En effet, cela implique plus de coalitions de grande cardinalité qui ont donc un biais collectif important. Ainsi, cela restreint les solutions collectivement stables aux coalitions de grande cardinalité et limite donc la qualité de l'exploration.

Les Figures 2.a, 2.b et 2.c montrent l'évolution de la somme des distances de Hellinger pour, respectivement, 5, 6 et 7 agents. Comme précédemment, les solutions individuellement stables sont représentées par les courbes bleues et les solutions collectivement stables sont représentées par les courbes rouges.

Ces résultats montrent clairement que construire des solutions individuellement stables peine davantage à réduire la distance entre les fonctions caractéristiques estimée et réelle en comparaison avec des solutions collectivement stables. Cependant, la différence entre les deux biais est variable. Pour 5, 6 et 7 agents, la différence est d'approximativement 8%, 24% et 14%. Elles peuvent s'expliquer notamment par le fait que le terme UCB du concept de solution collectivement stable possède davantage de poids car non divisé. Il est donc attendu que le biais collectif explore et donc apprenne davantage. Cet écart d'apprentissage est également lié à la rapidité à laquelle les biais découvrent les structures intéressantes. En effet pour 5 agents, nous pouvons voir que la courbe du biais individuel peine à se stabiliser tandis qu'à partir de 200 pas de temps la courbe du biais collectif se stabilise : l'écart se réduit donc. Pour 6 et 7 agents, le biais individuel n'explore plus beaucoup

passé 100 pas de temps, tandis que le biais collectif est encore en plein apprentissage.

Nous pouvons également noter que plus le nombre d'agents augmente, plus la somme des distances, et donc les erreurs d'apprentissage, augmentent. Pour 5, 6 et 7 agents, la somme des distances à 500 pas de temps est d'approximativement 0,62, 0,79 et 0,88 pour le biais individuel et de 0,53, 0,55 et 0,76 pour le biais collectif. Cette augmentation vient du fait que, malgré la division de la somme des distances par le nombre de coalitions, la quantité d'information à obtenir pour bien apprendre est proportionnellement plus importante à 7 agents qu'à 5, en raison du nombre croissant de coalitions pouvant être formées. Toutefois, notons que cette mesure indique la quantité d'information connue mais ne nous indique en rien si cette information est utile : il n'est pas utile aux agents de connaître avec précision la valeur d'une coalition qu'ils savent déjà être sans intérêt pour eux.

Il ressort de nos expérimentations que les solutions collectivement stables semblent plus performantes en ce qui concerne le regret, bien que cette tendance s'amenuise le nombre d'agents augmentant. Les solutions collectivement stables permettent également un apprentissage de la fonction caractéristique meilleur que les solutions individuellement stables. Cependant, ces dernières devenant beaucoup plus performantes quand le nombre d'agents tend à augmenter, nous pouvons en déduire qu'une meilleure connaissance de la fonction caractéristique ne permet pas nécessairement d'être plus performant en termes de regret. Cependant, suivant le nombre d'agents, le concept de solution et ses deux biais sont intéressants, permettant de former les structures de coalitions stables, et ce avec une connaissance partielle des fonctions caractéristiques, grâce à l'exploration.

5 Conclusion

Nous proposons dans cet article un modèle de jeux de coalitions répétés pour l'apprentissage d'une fonction caractéristique stochastique, a priori inconnue des agents. Nous avons introduit deux nouveaux concepts de solutions fondés sur celui de l' ϵ -cœur en l'adaptant à un contexte stochastique ainsi qu'en y ajoutant une notion d'exploration, celle-ci pouvant être individuelle ou collective. Les résultats montrent que les deux concepts de solutions sont intéressants, mais que de plus amples expérimentations s'imposent, notamment avec un nombre d'agents plus important.

Plusieurs pistes de recherche nous paraissent pertinentes.

D'une part, nous pouvons nous interroger sur la nature de la métrique de distance de Hellinger. En effet, bien que nous puissions comparer la fonction caractéristique estimée à la fonction réelle, cette métrique ne nous indique pas l'utilité de la fonction estimée. En effet, nous avons observé qu'une meilleure estimation ne conduisait pas nécessairement à un regret moindre. Moyennant des précautions au regard de notre critique du regret, cela traduit le fait que toutes les informations ne sont pas nécessaires pour pouvoir prendre de

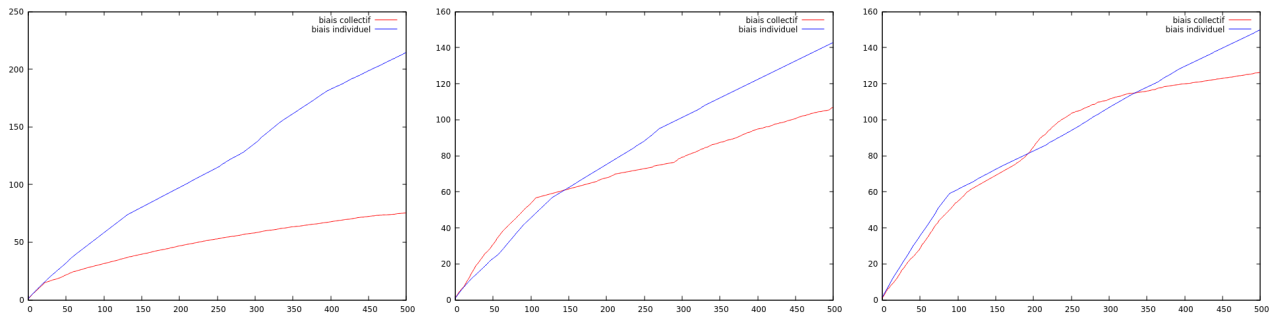


FIGURE 1 – De gauche à droite : regret cumulé collectif pour (a) 5 agents ; (b) 6 agents et (c) 7 agents

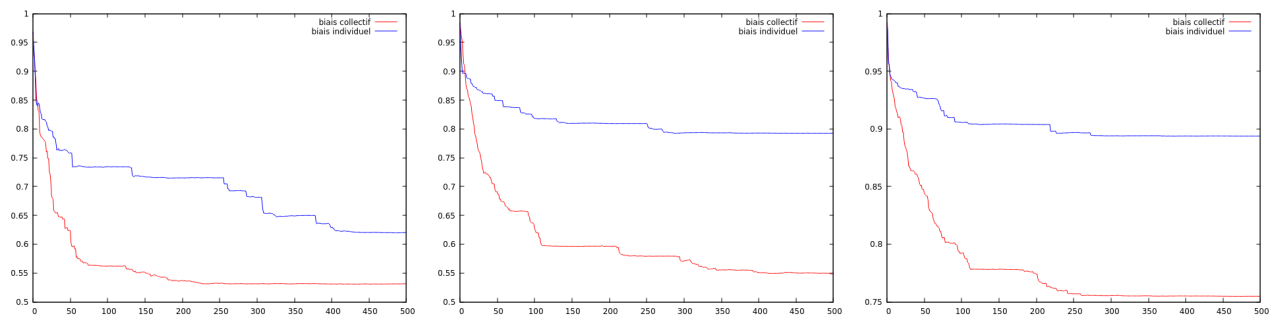


FIGURE 2 – De gauche à droite : somme des distances pour (a) 5 agents ; (b) 6 agents et (c) 7 agents

bonnes décisions. De plus, n’oublions pas que le regret est une mesure d’optimalité et non de stabilité. Il serait donc pertinent de considérer des métriques plus adaptées ou d’en développer de nouvelles.

D’autre part, le processus de formation de coalitions que nous avons considéré est un processus centralisé. Sans ce processus, les agents peuvent ne pas avoir la même estimation de la fonction caractéristique et vont devoir s’échanger des informations, se faire confiance à propos de ces dernières et décider collectivement en présence d’information asymétrique. Ainsi, nous pouvons nous demander comment les concepts de solutions collectivement et individuellement stables pourraient être transposés dans un tel contexte? Enfin, il pourrait être intéressant d’adapter d’autres concepts de solution à notre modèle ou encore s’intéresser de plus près à la répartition des gains, afin d’analyser comment celui-ci peut agir sur l’apprentissage et la décision.

Références

- [1] J.P. Benoit et V. Krishna. Finitely repeated games. *Econometrica*, Vol. 53, Issue 4, pages 905-922, 1985.
- [2] B. Blankenburg, R.K. Dash, S.D. Ramchurn, M. Klusch et N.R. Jennings. Trusted kernel-based coalition formation. *Proc. of 4th Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 989-996, 2005.
- [3] G. Chalkiadakis et C. Boutilier. Bayesian reinforcement learning for coalition formation under uncertainty. *Proc. of 3rd Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 1090-1097, 2004.
- [4] G. Chalkiadakis et C. Boutilier. Sequential decision making in repeated coalition formation under uncertainty. *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 347-354, 2008.
- [5] G. Chalkiadakis, E. Elkind et M. Wooldridge. Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Vol. 5, No. 6, pages 1-168, 2011.
- [6] G. Chalkiadakis, E. Markakis et C. Boutilier. Coalition formation under uncertainty : Bargaining equilibria and the Bayesian core stability concept. *Proc. of 6th Int. Conf. on Autonomous Agents and Multiagent Systems*, pages 400-407, 2007.
- [7] A. Charnes et D. Granot. Prior solutions : Extensions of convex nucleus solutions to chance-constrained games. *Texas Univ. Austin Center for Cybernetic Studies*, No. CS-118, 1973.
- [8] A. Charnes et D. Granot. Coalitional and chance-constrained solutions to n-person games. I : The prior satisficing nucleolus. *SIAM J. Appl. Math.*, Vol. 31, No. 2, pages 358-367, 1976.
- [9] R. Debraj. A game-theoretic perspective on coalition formation. *Oxford University Press*, 2007.
- [10] J.C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society : Series B (Methodological)*, Vol. 41, No. 2, pages 148-164, 1979.

- [11] S. Jeong et Y. Shoham. Bayesian coalitional games. *Proc. of 23rd AAAI Conf. on Artificial Intelligence*, pages 95-100, 2008.
- [12] A. Mahajan et D. Teneketzis. Multi-armed bandit problems. *Foundations and applications of sensor management*, Springer, Boston, MA, pages 121-151, 2008.
- [13] R. Mochaourab et E.A. Jorswieck. Coalitional games in MISO interference channels : Epsilon-core and coalition structure stable set. *IEEE Transactions on Signal Processing*, Vol. 62, No. 24, pages 6507-6520, 2014.
- [14] O. Morgenstern et J. Von Neumann. Theory of games and economic behavior. *Princeton University Press*, 1953.
- [15] T. Rahwan, T. Michalak, M. Wooldridge et N. R. Jennings. Anytime coalition structure generation in multi-agent systems with positive or negative externalities. *Artificial Intelligence*, Vol. 186, pages 95-122, 2012.
- [16] L.S. Shapley. A value for n -person games. *Contributions to the Theory of Games*, Vol. 2, No. 28, pages 307-317, 1953.
- [17] L. S. Shapley et M. Shubik. Quasi-cores in a monetary economy with non-convex preferences. *Econometrica*, Vol. 34, No. 4, pages 805-827, 1966

Problème d'affectation dynamique des emplacements de stockage chez Knapp : vers de l'apprentissage automatique ?

P. Courtin^{1,2}, A. Grimault¹, M. Lhommeau¹, J.-B. Fasquel¹

¹ Laboratoire Angevin de Recherche en Ingénierie des Systèmes (LARIS)
Université Angers, 62 avenue Notre Dame du Lac, 49000 Angers, FRANCE

² Knapp France, 23 Rue de la Maison Rouge, 77185 Lognes

paul.courtin@knapp.com

20 mars 2020

Résumé

L'optimisation des performances de préparation des entrepôts automatisés passe par une affectation judicieuse des produits aux emplacements de stockage. Ce problème d'affectation des positions de stockage (SLAP), est généralement abordé par les méthodes relevant de la recherche opérationnelle. Cet article dresse un état de l'art de cette problématique en soulignant les liens possibles avec l'apprentissage automatique, et des perspectives envisageables de résolution combinant apprentissage automatique et recherche opérationnelle.

Mots-clés

entrepôt automatisé, problème d'affectation des positions de stockage, affectation de position de stockage, apprentissage automatique, apprentissage par renforcement, affectation dynamique de position, SLAP, SBS/RS

Abstract

Automated warehouses picking performance optimization is strongly influenced by a wise assignment of product to storage location. The storage location assignment problem (SLAP), is usually solved with Operational Research methods. This article draws up a literature review to solve this problem by highlights link with Machine Learning. It also present plausible prospect to solve this problematic by combining Machine Learning and Operational Research

Keywords

automated warehouse, storage location assignment problem, storage allocation, machine learning, reinforcement learning, dynamic slotting, SLAP, SBS/RS

1 Introduction

La révolution numérique des entreprises, par la mise en place du paradigme *Industrie 4.0*, a apporté une profusion de données disponibles. L'exploitation de ces données peut constituer une aide à la prise de décision et à la pose de diagnostics, basée sur les données (approche *data-driven*). Néanmoins, il faut au préalable valoriser ces

données. Transformer les masses de données (*Big Data*) en informations (*Smart Data*) grâce, en outre, à l'utilisation d'algorithmes d'apprentissage automatique (*Machine Learning*) [2].

Dans ce contexte, nous nous intéressons aux problématiques d'optimisation de gestion d'entrepôts introduites par [9] et au coeur des préoccupations de la société Knapp. Ces problématiques traitent d'une manière générale de l'affectation d'articles à des conteneurs de stockage (*pallet assignment*), l'affectation de ces conteneurs à des emplacements de stockage (*storage assignment* ou *slotting*) et de l'ordonnancement des opérations de manutention (dépose/récupération) des conteneurs (*interleaving*). Dans notre étude, nous traiterons le problème d'affectation des conteneurs à des emplacements de stockage : *Storage Allocation Assignment Problem* (SLAP), souvent traité dans la littérature par des méthodes relevant de la recherche opérationnelle [9, 8, 21].

Comme ceci a été récemment souligné par [2], l'apprentissage automatique offre de multiples perspectives pour les applications industrielles (e.g. détection de pannes de composants, diagnostic de pannes, optimisation de production, détection de dommages). Néanmoins ce type d'approche est encore sous-exploité [15], malgré les perspectives offertes, dans de multiples domaines, par l'apprentissage profond (*Deep Learning*) [7], et l'apprentissage profond par renforcement (*Deep Reinforcement Learning*).

L'objectif de cet article est, pour notre contexte applicatif et ses contraintes propres, d'identifier les points sur lesquels l'apprentissage automatique pourrait s'avérer pertinent. Notre contribution consiste donc en la proposition d'une stratégie de gestion d'entrepôt permettant de traiter la problématique d'affectation des positions de stockage de manière dynamique, en s'appuyant sur les perspectives offertes par l'apprentissage automatique, combiné à la recherche opérationnelle. Cette proposition sera contextualisée en s'inspirant d'exemples d'applications connexes s'intéressant à l'apprentissage automatique.

La section 2 est dédiée à la présentation détaillée de la problématique et de l'état de l'art, en soulignant des dif-

ficultés encore peu traitées, liées au manque d'anticipation des commandes. La section 3 se focalise sur les méthodes de résolution du domaine, basées sur la recherche opérationnelle et sur l'apprentissage automatique. Enfin dans la section 4, nous proposons également des nouvelles pistes d'investigations.

2 Problématique et état de l'art

2.1 Description du système

Nous nous plaçons dans le contexte d'un entrepôt automatisé *goods-to-person* mettant en œuvre un transtockeur de type *Shuttle-based Storage and Retrieval Systems* (SBS/RS), qui est un dérivé du *Automated Storage and Retrieval System* (AS/RS) utilisant des navettes (*Shuttle*). Le transtockeur est composé d'une ou plusieurs allées. Au centre des rayonnages des allées, une navette évolue à chaque étage (*tier-captive shuttle*). Les navettes sont responsables du transport des conteneurs de stockage depuis et vers l'ascenseur (le point I/O) ainsi que de leur manutention dans les rayonnages (dépose et récupération). La figure 1 présente une vue de dessus d'un entrepôt. Le SBR/RS est configuré avec 4 allées, dont les rayonnages sont organisés en simple profondeur (*single deep*). Dans les rayonnages sont stockés les conteneurs de stockage contenant les unités de gestion du stock (*Stock Keeping Unit* ou SKU).

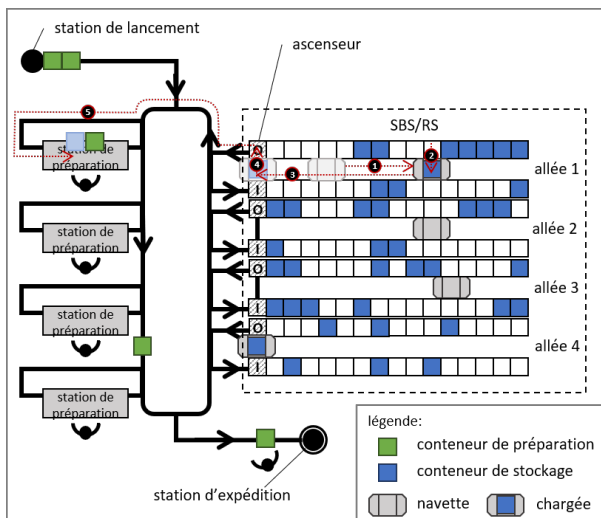


FIGURE 1 – Schéma d'un entrepôt avec un SBS/RS

La figure 1 illustre l'environnement associé à notre problématique. Dans un entrepôt logistique, plusieurs opérations sont nécessaires pour assurer la préparation des commandes :

1. *Goods-in* : étape d'entrée des produits dans le système. Au préalable, les SKU doivent être disponibles pour la préparation, i.e. stockés dans le SBS/RS.
2. *Picking* : étape de préparation. Les conteneurs de préparation sont acheminés depuis une station de lancement vers les stations de préparation, et les

conteneurs de stockage (contenant le stock) sont acheminés vers les mêmes stations de préparation (étapes 1,2,3 et 4 de la figure 1). Une fois les conteneurs de préparation et de stockage arrivés en station de préparation, les préparateurs peuvent alors transférer les produits des conteneurs de stockage vers les conteneurs de préparation.

3. *Shipping* : étape d'expédition. Le *picking* terminé, le conteneur de préparation est acheminé vers l'expédition et le conteneur de stockage retourne vers le SBS/RS pour être de nouveau stocké.
4. *Slotting* : étape d'affectation. Le système de gestion d'entrepôt *Warehouse Management System* (WMS) va sélectionner la meilleure position de stockage pour le conteneur de stockage, en fonction des caractéristiques des SKU présents dans le conteneur, de l'historique des commandes et de la stratégie de stockage implémentée.

2.2 Optimisation d'un entrepôt automatisé

Dans l'optique d'obtenir ou conserver un avantage concurrentiel, les industriels s'efforcent d'améliorer les performances de leurs entrepôts par une meilleure conception et l'optimisation de la conduite des opérations. L'optimisation d'un entrepôt automatisé peut être définie par plusieurs objectifs concomitants à minimiser : la distance moyenne de trajet d'un circuit de préparation, les coûts d'investissements et d'exploitation, le temps de préparation d'une commande ou d'un groupe de commandes, l'utilisation des équipements et des personnes, ou la perte d'espace [12]. Atteindre ces objectifs requière de prendre des décisions, que l'on peut hiérarchiser en niveaux stratégique, tactique et opérationnel [22]. Ces décisions portent sur différents points : conception des plans de l'entrepôt et dimensionnement (*layout design*), affectation des produits à des emplacements de stockage (*storage assignment*), affectation des commandes à des tournées et regroupement des allées en zone de travail (*batching and zoning*), routage des préparateurs de commande (*routing*), triage et accumulation des commandes (*order sorting and accumulation*) [12]. Comme ceci est illustré par la figure 2, les opérations de préparation de commande ont été identifiées [12] comme étant l'activité la plus coûteuse en temps et nécessitant une main d'œuvre conséquente.

55% des coûts opérationnels sont imputables à la préparation des commandes [27]. Le temps de préparation est lui-même impacté par (i) la durée de trajet (*Travel*) des SKU vers, depuis et entre les positions de préparations (à hauteur de 50%), (ii) la durée d'identification (*Search*) des SKU nécessaires à la préparation des commandes (à hauteur de 20%), (iii) la durée de prélèvement (*Picking*) des SKU depuis les conteneurs de stockage (à hauteur de 15%), (iv) la durée de préparation (*Setup*) de la tournée (à hauteur de 10%) et (v) d'autres tâches (*Other*) (à hauteur de 5%) [23], comme illustré dans la figure 2. Pour cette raison, nous nous focalisons sur la réduction du temps de préparation de commandes, et plus particulièrement sur la minimisation du temps (ou distance) de trajet des navettes du

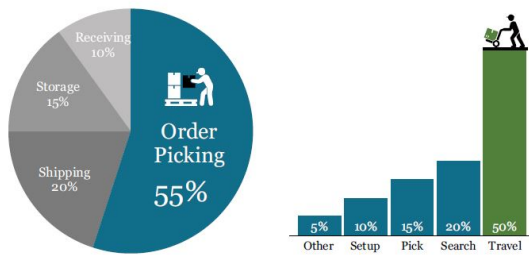


FIGURE 2 – Répartition des coûts opérations logistiques cumulées sur une année dans un entrepôt (gauche) et répartition du temps de chaque tâche sur l'activité de préparation de commandes (*order picking* - droite) (extrait de [11])

SBS/RS. Dans cette optique, nous allons étudier l'affectation des produits à des emplacements de stockage optimaux pour réduire le coût des opérations de préparation.

L'affectation de produits à des positions de stockage est connue dans la littérature comme étant le *Storage Location Assignment problem* (SLAP), dont la première formulation a été introduite par [9], et a été prouvé NP-hard par [5]. Le SLAP peut être abordé de différentes manières [8] selon la nature des données d'entrées : dates d'entrée et de sortie du produit connues dans le système, caractéristiques des produits (poids, dimensions), etc. Lors de l'affectation des conteneurs (et donc des produits qu'ils contiennent) à des emplacements de stockage (*Slotting*), le WMS s'efforce de choisir des emplacements en respectant une stratégie de stockage qui permet d'obtenir de bonnes performances lors du *Picking* tout en respectant des règles métiers et opérationnelles :

- la masse globale des produits doit être répartie sur toute la longueur du rayonnage pour limiter les contraintes mécaniques ;
- des espaces vides doivent être laissés pour faciliter les opérations de manutention intra-allées ;
- les produits comburants et combustibles ne doivent pas être stockés les uns à côté des autres ;
- le stock doit être distribué entre les allées et les étages pour ne pas bloquer le stock en cas d'indisponibilité d'un appareil (ascenseur ou navette).

Ces règles sont autant de contraintes à prendre en compte pour résoudre le SLAP. Mais, elles sont peu considérées dans les formalisations des problèmes, notamment la répartition de la masse, comme souligné dans l'état de l'art de [21]. Pour résoudre le SLAP, des méthodes exactes (programmation dynamique, programmation linéaire mixte en nombre entiers) et des méthodes approchées (heuristique et métaheuristiques) ont été proposées [26]. Plusieurs stratégies de stockage peuvent être envisagées, induisant des règles et une formulation particulières pour traiter le problème de SLAP [11]. Les stratégies les plus courantes sont :

- *Class-Based* (CB) : les produits sont séparés en plusieurs classes et une zone de stockage est affectée à chaque classe. La plus populaire est l'affec-

tation en 3 classes, ABC où les SKU sont répartis suivant un critère tel que le *turnover* (e.g. les produits les plus fréquemment commandés sont placés dans la classe A, associée à la zone la plus proche du point I/O), introduite par [9]. À l'intérieur de chaque classe, les produits sont rangés avec une règle simple (e.g. position disponible la plus proche du point I/O). D'autres classifications peuvent être envisagées. Par exemple, la classification XYZ (détaillée dans la section 2.3) s'appuie sur les fluctuations de commandes d'un produit [19, 24] ;

- *Duration-of-Stay* (DoS) : les produits sont affectés à un emplacement dont la distance depuis le point I/O est proportionnelle à la durée de séjour dans l'entrepôt. Sous certaines conditions (DoS connu à l'avance et nombre équilibré d'entrées/sorties de marchandises), cette règle est optimale [6] (par rapport aux autres stratégies, en terme de temps de trajet et d'occupation d'espace) ;
- *Cube-Order-per-Index* (COI) : cette métrique définit le ratio entre l'espace nécessaire pour stocker un produit et sa demande [6]. Les produits avec un faible COI (i.e. de petite taille et souvent demandé) sont placés près du point I/O ;
- *Random-Based* : cette politique affecte les produits à des emplacements de manière aléatoire. Elle peut être utilisée dans l'industrie et est considérée dans la littérature comme base de comparaison.

2.3 Problème de la fluctuation des commandes

Dans notre contexte opérationnel et industriel, au-delà des contraintes métiers présentées précédemment (répartition de la masse, espaces vides, compatibilité entre produits et distribution spatiale), des complications supplémentaires ont été détectées, lors de l'exploitation des algorithmes de *Slotting* (implémentés dans les WMS). Ces complications résultent d'aspects qui sont souvent ignorés :

1. la fluctuation de la fréquence de demande et de la quantité des SKU dans les commandes ;
2. la survenue d'événements rares mais significatifs [1]. Par exemple, il peut s'agir de promotions conduisant à un pic de commande soudain et bref.

Ce dernier point est illustré par la figure 3 qui représente un historique des commandes observées par l'entreprise Knapp, dans un entrepôt donné et sur une période spécifique. La première ordonnée (à gauche) représente la quantité de pièces commandées en fonction de la semaine sur une période d'environ 6 mois. La seconde ordonnée (à droite) représente la performance de préparation, exprimée (par convention) en nombre de prélèvements par heure (*pick/h* - moyenne sur une semaine). Dans notre exemple, on constate une baisse de la performance de préparation, d'environ 30% en semaine 10. Cette chute de la performance pourrait s'expliquer par une réduction de la quantité de travail (nombre pièces commandées) en semaine 10, ce qui impliquerait une chute de la performance de prépa-

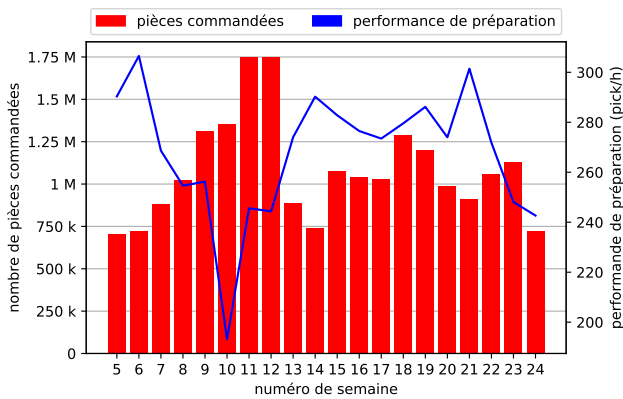


FIGURE 3 – Nombre de pièces commandées et performance de *picking* en fonction de la semaine : un exemple.

ration. Mais, on constate que la quantité totale de SKU à préparer augmente de +3% entre les semaines 9 et 10. Une variation aussi faible du nombre de pièces commandées ne devrait pas impacter autant la performance. La chute de performance résulte d'un pic de commande sur un type particulier de pièce, détaillé ci-après.

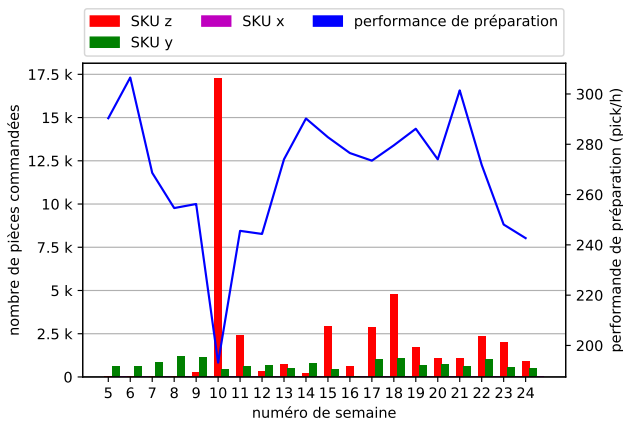


FIGURE 4 – Nombre de pièces commandées pour des SKU étiquetés X, Y et Z et la performance de *picking* en fonction de la semaine.

La figure 4 représente le nombre de pièces commandées pour 3 SKU en fonction de la semaine et la seconde ordonnée représente la performance de préparation. Les 3 SKU choisis sont représentatifs des types variations de demande observées dans les entrepôts. Le produit étiqueté x a une demande faible voire négligeable (moins de 0,005% de la demande totale), avec une faible variation dans le temps. Le produit étiqueté y se place à un niveau de demande plus important (près de 0,12%) et on observe une légère variation de quantité dans le temps. Enfin, le produit étiqueté z a une variation de demande brutale. On constate que ce produit, jamais commandé avant la semaine 10, devient en semaine 10 un produit très demandé (17 000 pièces sur un total de 1 352 273 requises soit près de 1,28% de la

demande totale) et ce pour une durée de seulement 1 semaine. Ensuite sa demande devient sporadique, mais reste significative (moins de 0,37% de la demande).

Ces comportements de variation de demande sont identifiés dans la littérature grâce à la classification par analyse XYZ [19, 24]. La classe X regroupe les produits dont la variation de demande (dans le temps) est stable, uniforme, et continue avec de faible variation. La classe Y regroupe les produits dont la demande fluctue davantage, notamment avec la saison. Enfin, la classe Z regroupe les produits dont la demande est sporadique, irrégulière voir anormale en terme de quantité par rapport aux autres produits. Les variations demande des produits Z ne sont peu ou pas prédictibles.

Cette classification des SKU s'applique à notre exemple, le produit étiqueté $x \in X$, celui étiqueté $y \in Y$ et étiqueté $z \in Z$. Le produit étiqueté z , a subi une variation soudaine et brève du nombre de produits demandés. Il a donc été identifié par le WMS comme un produit à forte rotation. Par conséquent ce SKU s'est vu assigner les emplacements de stockage au plus près du point I/O. Le fonctionnement du WMS est parfaitement normal, sur la base des données dont il dispose (historique des commandes, variation de demande passé des SKU) et de la stratégie de stockage utilisé. L'algorithme de *slotting* a assigné un produit à forte demande sur les emplacements de stockage dont l'accès est le plus "rapide". Le problème est néanmoins le suivant : cette explosion de commandes pour la référence z est valide pour une courte et brève période. Les semaines suivantes ne requièrent que peu de cette référence de produit, qui se retrouve dans le dernier quartile des produits demandés. Toutefois, le SKU étiqueté Z occupe les "meilleurs" emplacements de stockage, ce qui implique des déplacements plus long pour les navettes, qui doivent dépasser cette zone de stockage, pour atteindre les autres SKU.

Le problème illustré par cet exemple est : comment exploiter l'historique de commandes pour positionner les SKU dans l'entrepôt de manière optimale? À noter que cette question pose le problème sous-jacent suivant : faut-il recalculer les affectations de positions pour chaque SKU et donc modifier dynamiquement l'agencement de l'entrepôt? Ceci impliquerait des mouvements de manutention intra-allées pour repositionner les conteneurs et du temps d'indisponibilité du stock et des navettes (occupées à des tâches sans valeur ajoutée immédiate). Ceci sous-entend de définir une politique de gestion de l'entrepôt permettant d'effectuer cette manutention sans perturber la production. À notre connaissance, cet aspect n'est que trop peu abordé dans la littérature. On parle de *Dynamic Storage Location Assignment Problem* (DSLAP). Ce problème a été abordé par des méthodes heuristiques [8]. Concernant l'exploitation de l'historique des commandes en particulier, et dans le cas d'une stratégie CB (ABC ou XYZ), l'industrie s'appuie essentiellement sur une analyse superficielle des données (moyenne des commandes passées pour ABC ou leur fluctuation moyenne pour XYZ), sans chercher à modifier l'agencement de l'entrepôt dynamiquement.

La section suivante se focalise sur une modélisation plus précise de cette problématique et des liens possibles entre

la recherche opérationnelle et l'apprentissage automatique.

3 Modélisation, recherche opérationnelle et apprentissage automatique

Compte tenu de la problématique décrite précédemment, incluant la fluctuation des commandes, nous proposons ici des pistes concernant la modélisation de ce problème d'optimisation et son couplage avec l'apprentissage automatique, en s'appuyant sur les travaux récemment publiés sur ce sujet. La section 3.1 est dédiée à la formulation du problème d'optimisation du SLAP. La section 3.2 concerne l'usage et les perspectives de l'apprentissage automatique dans notre contexte. Nous nous intéresserons en particulier à l'apprentissage non supervisé, l'apprentissage supervisé et l'apprentissage par renforcement, en soulignant les liens avec l'apprentissage par réseaux de neurones profonds.

3.1 Modélisation du problème d'optimisation

Dans cette partie, nous nous intéressons à une formulation de notre problème d'optimisation. Les formulations du SLAP sont basées sur le problème *General Assignment Problem* (GAP) [10].

Les entités souvent rencontrées dans la littérature sont les suivantes. Soit $\mathcal{N} = \{1, \dots, N\}$ l'ensemble des SKU (éléments) à stocker. Soit $\mathcal{L} = \{1, \dots, L\}$ l'ensemble des positions. Soit $\mathcal{O} = \{1, \dots, O\}$ l'ensemble des commandes. Nous définissons une variable binaire x_{ik} valant 1 si l'article de type $i \in \mathcal{N}$ est assigné à la position $k \in \mathcal{L}$, 0 sinon. Dans chacun des rayonnages des allées, la distance entre une position de stockage k et le point d'entrée/sortie I/O, situé au niveau de l'ascenseur, est notée $d_k \geq 0, k \in \mathcal{L}$. La séquence de sortie des SKU et la quantité requise sont données par l'ensemble des commandes de préparation $\mathcal{O} = \{1, \dots, O\}$. Cet ensemble permet de déterminer la probabilité de demande de chaque SKU i , notée $p_i \geq 0, i \in \mathcal{N}$. À noter que cette probabilité repose généralement sur une mesure simpliste (nombre moyen de commandes par article), et ne permet pas de détecter les pics si l'intervalle d'estimation est trop important.

Une liste de propriétés peut être ajoutée à chaque article appartenant à l'ensemble \mathcal{N} : classes d'éléments compatibles (e.g. problématique des produits inflammables), poids, dimensions, etc.

Il existe plusieurs manières de formuler le problème, selon la modélisation de l'entrepôt et le (ou les) critère(s) que l'on cherche à optimiser (e.g. voir les récents travaux intégrant une optimisation multi-objectif [28, 4]). Nous donnons ici un exemple simplifié, inspiré de [29] dans le cas où le critère d'optimisation est de minimiser la distance total de trajet des navettes et la répartition du poids. Pour cela, nous définissons w_i , correspondant à la masse du SKU i , et l'entité $\mathcal{M} = \{1, \dots, M\}$ l'ensemble des colonnes (voir la figure 5). Une colonne m_i est une partition de l'ensemble \mathcal{L} .

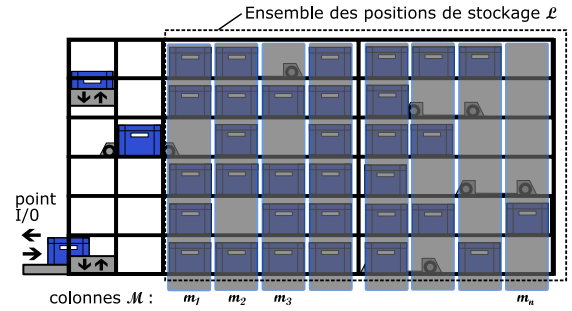


FIGURE 5 – Schéma explicatif des colonnes du SBS/RS considérées pour la contrainte de répartition de masse

La fonction objectif à optimiser peut être formulée par z :

$$z = \min \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{L}} x_{ik} \cdot d_k \cdot p_i \quad (1)$$

$$\sum_{i \in \mathcal{N}} x_{ik} \leq 1 \quad \forall k \in \mathcal{L} \quad (2)$$

$$\sum_{k \in \mathcal{L}} x_{ik} = 1 \quad \forall i \in \mathcal{N} \quad (3)$$

$$\left| \sum_{i|k \in m_1} w_i x_{ik} - \sum_{i|k \in m_2} w_i x_{ik} \right| \leq \varepsilon \quad \forall (m_1, m_2) \in M^2$$

$$m_1 \neq m_2 \quad (4)$$

$$x_{ik} \in \{0, 1\} \quad \forall i \in \mathcal{N}, \forall k \in \mathcal{L}$$

Les contraintes (2) imposent un seul produit pour chaque position, les contraintes (3) imposent une seule position pour chaque produit. Les contraintes (4) imposent de répartir la masse globale des SKU sur la totalité de la longueur du rayonnage (ε étant un paramètre associé aux écarts de poids entre colonnes). On pourrait étendre ces contraintes en s'appuyant sur d'autres propriétés associées aux SKU, par exemple le problème de compatibilité entre les produits (produits inflammables).

Cette formulation est assez similaire à la stratégie DoS (où notre p_i peut être associé au temps de présence dans le système), en lien avec de l'apprentissage profond [15].

Dans le cas d'une stratégie de type *Class-Based*, on pourrait classifier les produits en catégorie et adapter la formulation en intégrant une nouvelle entité $\mathcal{C} = \{1, \dots, C\}$, désignant l'ensemble des classes de stockage définies : les produits sont alors pré-affectés à des zones, induisant des contraintes supplémentaires sur les positions d'affectation (variables x_{ik}).

3.2 Apprentissage automatique

Comme souligné dans l'introduction, l'apprentissage automatique et en particulier l'apprentissage profond ont montré leur efficacité dans de nombreux domaines [7]. Cela fait notamment partie des perspectives récemment formulées concernant l'usage de l'apprentissage profond pour la prédiction des commandes et la catégorisation des produits [31, 21].

Plusieurs approches peuvent être envisagées pour exploiter l'historique des commandes, en s'appuyant sur l'apprentissage automatique. Nous allons considérer ci-après trois approches suscitant un intérêt récent de la communauté scientifique : l'apprentissage non-supervisé, supervisé et l'apprentissage par renforcement. Dans chaque cas, nous donnons quelques récents travaux, en soulignant les éventuels liens avec l'apprentissage profond, autour de notre problématique et autour de problématiques connexes, utilisant ces approches, soulignant ainsi leur potentiel pour traiter notre sujet.

3.2.1 Apprentissage non supervisé

L'apprentissage non supervisé permet, sur la base des caractéristiques des objets manipulés ou de leurs interactions, d'identifier des groupes d'objets présentant des caractéristiques similaires.

Dans notre cas, l'apprentissage non supervisé se retrouve sur le SLAP pour classer les SKU en catégories, lorsque la stratégie de stockage est de type *Class-Based*. Les SKU sont répartis dans des classes \mathcal{C} , mentionnées dans la section précédente, grâce à l'apprentissage automatique [30]. Les classes de produits peuvent être établies par des techniques de *clustering* s'appuyant sur les attributs des différents SKU commandés et l'historique des commandes. Dans l'étude du SLAP d'un entrepôt de stockage par *clustering* [30] dans le e-commerce, se base sur des attributs standard de la classification *Class-Based*, tels que le prix et le volumes des ventes, auxquels il associe des attributs complémentaires : le taux de commentaires favorables des SKU, le type de paiement autorisé à la livraison (paiement en espèce ou non), si les frais de port sont offerts ou non. Il a pu ainsi guider le processus de *clustering*, et répartir 109 SKU en 3 catégories.

Cette méthode de *clustering* est basée sur les attributs intrinsèques des SKU, d'autres méthodes de classification se basent sur les interactions entre les SKU. Des règles d'associations *association rule mining* permettent d'extraire des relations d'intérêts entre les SKU. Ces règles fournissent alors des métriques (e.g *weighted support count* (WSC) [18], *product affinity* [14]), qui sont utilisées pour catégoriser les SKU et les assigner aux positions de stockage.

Les méthodes de classification étudiées font peu appel aux réseaux de neurones. Or les méthodes de classification par apprentissage profond font parties des perspectives récemment formulées pour la catégorisation des produits [31, 21].

3.2.2 Apprentissage supervisé

L'apprentissage supervisé a pour objectif la prédiction et/ou la classification, sur la base d'un entraînement réalisé sur des données annotées.

L'apprentissage supervisé peut également se retrouver dans la problématique du SLAP, par exemple dans le cas d'une stratégie de type *Duration of Stay* [15]. Ces travaux introduisent le terme \hat{p} désignant la durée de séjour (prédite) d'un SKU dans le système, le *Duration of Stay* (DoS), qui est estimé avec un réseau de neurones profond convolutif

(*convolutional neural networks* ou CNN) et un réseau de neurones récurrent (*recurrent neural networks* ou RNN) de type *long short-term memory* (LSTM) [3]). Ce réseau est entraîné sur un historique de commandes, correspondant à l'entité \mathcal{O} introduite précédemment. Pour un conteneur avec un DoS prédit par apprentissage automatique, noté \hat{p} , la fonction d'affectation des positions de stockage $A : \mathbb{R} \rightarrow \mathcal{W}$ est, selon [15] :

$$A(\hat{p}) = \arg \min_{w \in \tilde{\mathcal{W}}} d(N\hat{r}\hat{W}(\hat{p}), w) + c(w) \quad (5)$$

Avec $d(v, w)$ la distance entre deux positions v et w . $c(w)$ représente les coûts connexes associés au stockage à cette position w . Cela prend en compte les coûts des contraintes spécifiques de chaque entrepôt, (e.g : *stock-mixing*, *stock-splitting*, répartition de la masse,...). La position de stockage optimale approchée dans le cas du DoS est donnée par $N\hat{r}\hat{W}(\hat{p})$. $\tilde{\mathcal{W}}$ représente l'ensemble des positions de stockage disponibles lorsqu'un conteneur de stockage entre dans l'entrepôt. $d(N\hat{r}\hat{W}(\hat{p}), w)$ représente le coût de non stockage d'une palette à la position de stockage optimale pour le DoS. La position de stockage $A(\hat{p})$ est alors choisie en fonction de ces deux coûts (les coûts connexes $c(w)$ et le coût de non stockage $d(N\hat{r}\hat{W}(\hat{p}), w)$) [15].

Le principe de cette approche est illustrée par la figure 6, dans le cas présent (DoS) et dans la perspective d'une utilisation dans le cas d'une stratégie de type *class-based*. On peut noter ici que cette approche combine apprentissage profond et optimisation sous contraintes.

À notre connaissance, l'article de [15] constitue le premier article qui traite du SLAP avec des réseaux de neurones profonds, en prenant en compte le caractère dynamique de l'entrepôt et les incertitudes d'exploitation. À noter que ces travaux sont associés à la mise à disposition de l'une des seules bases de données publiques contenant des données d'entrepôt pour traiter le problème d'affectation.

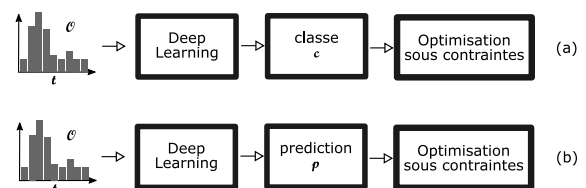


FIGURE 6 – Apprentissage profond et Optimisation par contraintes. Variante (a) les réseaux de neurones retournent la classe de stockage c pour les SKU. Variante (b) les réseaux de neurones prédisent la valeur \hat{p} du DoS pour les SKU

La figure 6 présente deux axes de recherche envisagés pour résoudre le SLAP. La variante (a) représente l'utilisation de *Deep Learning* pour catégoriser les produits en classes c_i en se basant sur l'historique des commandes, inspiré de [30, 14, 18]. La variante (b), correspond aux travaux de [15], utilise du *Deep Learning* pour prédire la valeur d'une métrique (le DoS pour [15], la probabilité de demande des

SKU dans notre cas). Dans les deux variantes (a) et (b), des techniques de *Deep Learning* vont transformer/utiliser les données, ici l'historique des commandes \mathcal{O} , pour fournir des variables d'entrée pour résoudre notre problème d'optimisation sous contrainte.

3.2.3 L'apprentissage par renforcement : prendre en compte le long terme

L'apprentissage par renforcement [25] est une autre piste intéressante n'ayant, à notre connaissance, pas encore été étudiée dans notre contexte.

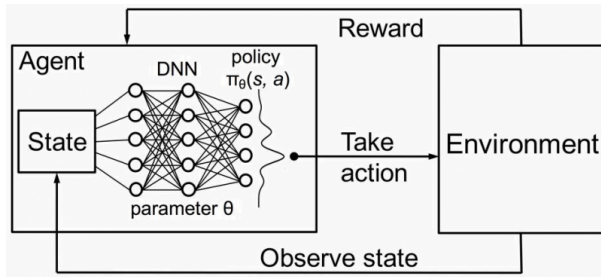


FIGURE 7 – *Deep reinforcement learning* (extrait de [17])

Le principe est illustré par la figure 7. Néanmoins, cette classe d'approche fait actuellement l'objet de travaux pour des problèmes connexes. On peut citer la répartition en charge de l'information [16] (apprentissage pour l'affectation temps réel des tâches sur des serveurs distribués), la replanification de tâches (i.e. des commandes passées par les clients) dans les environnements de production [20], la gestion des ressources en énergie en fonction des besoins (avec incertitude sur la demande) [13]. Elle repose sur une observation de l'environnement afin d'identifier les actions à entreprendre afin de maximiser les récompenses sur moyen ou long terme. s correspond à l'état observé et a à l'action optimale à entreprendre, sur la figure 7 l'action optimale est donnée par $\operatorname{argmax}_a(\pi_\theta(s, a))$ (θ est déterminé par la phase d'entraînement du réseau de neurones). Pour notre étude, une difficulté est de correctement modéliser le système et de définir les rôles : environnement, agent, action (e.g. réagencement de l'entrepôt dynamiquement) et récompenses (e.g. minimiser, sur un ensemble de commandes, le temps ou la distance de trajet des navettes de l'SBS/RS pour récupérer des SKU).

Dans notre cas, les différents rôles pourraient être attribués de la manière suivante :

- l'**environnement** est constitué du SBS/RS ;
- l'**agent** engage l'action optimale, apprise par renforcement ;
- l'**action** consiste à stocker le SKU $i \in \mathcal{N}$ à la position $k \in \mathcal{L}$;
- l'**état** du système est constitué de l'état d'occupation des positions de stockage \mathcal{L} , l'ensemble des commandes courantes, l'ensemble des SKU à stocker \mathcal{N} et de l'ensemble des classes de stockage \mathcal{C} dans le cas d'une stratégie *class-based* ;
- la **récompense** (*reward*) est associée au bénéfice

obtenu suite à l'action entreprise (associée à la baisse de la distance moyenne des trajets des navettes par exemple).

4 Conclusion

Nous avons vu que l'optimisation d'un entrepôt automatisé utilisant un SBS/RS, passe par la réduction du temps de préparation et donc par la réduction du temps/distance de trajet des navettes. Améliorer l'affectation des positions de stockage (*Slotting*) est un moyen d'action pour minimiser les trajets des navettes, connu dans la littérature comme le *Storage Location Assignment Problem* (SLAP). Nous avons vu, à travers un exemple concret, certaines difficultés associées à la fluctuation, difficilement prévisible, des commandes. Ceci requiert de tendre vers un système d'allocation dynamique basé sur l'apprentissage de la dynamique des commandes passées.

Nous avons ensuite présenté les approches considérées pour aborder cette problématique, généralement formulée comme un problème d'optimisation sous contraintes, relevant de la recherche opérationnelle. Nous avons également fait l'état de l'art des travaux connexes impliquant de l'apprentissage automatique. Cette étude des récents travaux publiés dans la littérature montre le potentiel de l'apprentissage automatique pour ce problème, et en particulier de l'apprentissage supervisé par réseaux de neurones profonds et également de l'apprentissage par renforcement. Cette étude montre également que peu de travaux ont été réalisés sur l'usage de l'apprentissage profond dans ce contexte. Enfin, ce travail de synthèse illustre comment l'apprentissage profond peut-être avantageusement combiné à la modélisation du problème, posé sous forme d'un problème d'optimisation de contraintes : l'apprentissage peut permettre de modéliser les commandes afin de guider le processus d'optimisation. La suite à moyen terme de ces travaux concernera l'approfondissement de cette étude et la réalisation d'un simulateur avant l'évaluation sur un environnement réel.

Références

- [1] Kamil Ciosek and Shimon Whiteson. Off-Environment RL with Rare Events. *NIPS workshop on Optimizing the Optimizers*, 2016.
- [2] Alberto Diez-Olivan, Javier Del Ser, Diego Galar, and Basilio Sierra. Data fusion and machine learning for industrial prognosis : Trends and perspectives towards industry 4.0. *Information Fusion*, 50 :92 – 111, 2019.
- [3] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description, 2014.
- [4] Marcele Elisa Fontana and Vilmar Santos Nepomuceno. Multi-criteria approach for products classification and their storage location assignment. *The Inter-*

- national Journal of Advanced Manufacturing Technology*, 88(9) :3205–3216, Feb 2017.
- [5] Edward H. Frazelle. *Stock location assignment and order picking productivity*. PhD thesis, Georgia Institute of Technology, 1989.
- [6] Marc Goetschalckx and H. Donald Ratliff. Shared storage policies based on the duration stay of unit loads. *Management Science*, 36(9) :1120–1132, 1990.
- [7] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.
- [8] Jinxiang Gu, Marc Goetschalckx, and Leon F. McGinnis. Research on warehouse operation : A comprehensive review. *European Journal of Operational Research*, 177(1) :1–21, 2007.
- [9] Warren Hausman, Leroy Schwarz, and Stephen Graves. Optimal Storage Assignment in Automatic Warehousing Systems. *Management Science*, 22(6), 1976.
- [10] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, Berlin, Germany, 2004.
- [11] Monika Kofler. *Optimising the Storage Location Assignment Problem Under Dynamic Conditions*. PhD thesis, Johannes Kepler Universität Linz, 2014.
- [12] René de Koster, Tho Le-Duc, and Kees Jan Roodbergen. Design and control of warehouse order picking : A literature review. *European Journal of Operational Research*, 182(2) :481 – 501, 2007.
- [13] Jesus Lago, Ecem Sogancioglu, Gowri Suryanarayana, Fjo De Ridder, and Bart De Schutter. Building day-ahead bidding functions for seasonal storage systems : A reinforcement learning approach. *IFAC PapersOnLine*, 52 :488–493, 06 2019.
- [14] Jiayi Li, Mohsen Moghaddam, and Shimon Y. Nof. Dynamic storage assignment with product affinity and abc classification—a case study. *The International Journal of Advanced Manufacturing Technology*, 84(9) :2179–2194, Jun 2016.
- [15] Michael Lingzhi Li, Elliott Wolf, and Daniel Wintz. Duration-of-stay storage assignment under uncertainty, 2019.
- [16] Long Mai, Nhu-Ngoc Dao, and Minho Park. Real-time task assignment approach leveraging reinforcement learning with evolution strategies for long-term latency minimization in fog computing. *Sensors*, 18(9), 2018.
- [17] Hongzi Mao, Mohammad Alizadeh, Ishai Menache, and Srikanth Kandula. Resource management with deep reinforcement learning. In *Proceedings of the 15th ACM Workshop on Hot Topics in Networks - HotNets '16*, pages 50–56. ACM Press, 2016.
- [18] David Ming-Huang Chiang, Chia-Ping Lin, and Mu-Chen Chen. Data mining based storage assignment heuristics for travel distance reduction. *Expert Systems*, 31(1) :81–90, 2014.
- [19] Irena Nowotyńska. An application of XYZ analysis in company stock management. *Modern Management Review*, 2013.
- [20] Jorge A. Palombarini and Ernesto C. Martínez. Closed-loop rescheduling using deep reinforcement learning. *IFAC-PapersOnLine*, 52(1) :231 – 236, 2019.
- [21] J. Reyes, E. Solano-Charris, and J. Montoya-Torres. The storage location assignment problem : A literature review. *International Journal of Industrial Engineering Computations*, 10 :199–224, 2019.
- [22] B. Rouwenhorst, B. Reuter, V. Stockrahm, G.J. van Houtum, R.J. Mantel, and W.H.M. Zijm. Warehouse design and control : Framework and literature review. *European Journal of Operational Research*, 122(3) :515–533, 2000.
- [23] André Scholz, Sebastian Henn, Meike Stuhlmann, and Gerhard Wäscher. A new mathematical programming formulation for the single-picker routing problem. *European Journal of Operational Research*, 253(1) :68–84, 2016.
- [24] Milan Stojanović and Dušan Regodić. The significance of the integrated multicriteria ABC-XYZ method for the inventory management process. *Acta Polytechnica Hungarica*, 14(5) :20, 2017.
- [25] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning : An Introduction*. The MIT Press, second edition, 1998.
- [26] El-Ghazali Talbi. Combining metaheuristics with mathematical programming, constraint programming and machine learning. *Annals of Operations Research*, 240(1) :171–215, 2016.
- [27] James A Tompkins, John A White, Yavuz A. Bozer, and J.M.A Tanchoco. *Facilities planning*. John Wiley & Sons, 4 edition, 2010.
- [28] Wei Wang, Jing Yang, Li Huang, David Proverbs, and Jianbin Wei. Intelligent storage location allocation with multiple objectives for flood control materials. *Water*, 11(8), 2019.
- [29] Ren-Qian Zhang, Meng Wang, and Xing Pan. New model of the storage location assignment problem considering demand correlation pattern. *Computers and Industrial Engineering*, 129 :210 – 219, 2019.
- [30] Li Zhou, Lili Sun, Zhaochan Li, Weipeng Li, Ning Cao, and Russell Higgs. Study on a storage location strategy based on clustering and association algorithms. *Soft Computing*, 2018.
- [31] E. Zunic, H. Hasic, K. Hodzic, S. Delalic, and A. Besirevic. Predictive analysis based approach for optimal warehouse product positioning. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 0950–0954, 2018.

Group-based Ordinal Classification based on a Negotiation Process

Ons Nefla^{1,2}, Imène Brigui², Meltem Öztürk¹, Paolo Viappiani³, Oussama Raboun.⁴

¹ Université de Paris Dauphine, PSL Research University CNRS, LAMSADE, 75016 Paris, France

² Emlyon business school, 23 Avenue Guy de Collongue, Ecully, France

³ Sorbonne Université, CNRS, Laboratoire d'informatique de Paris 6, LIP6, F-75005 Paris, France

⁴ Keley Consulting, 28 Rue du Dr Finlay, 75015 Paris, France

ons.nefla@dauphine.eu, brigui-ctiouyi@em-lyon.com, meltem.ozturk@dauphine.fr,
paolo.viappiani@lip6.fr, oraboun@keley-data.com

Abstract

In group decision making problems, reaching a common ordinal classification satisfying fairly each of the group members is a highly challenging objective. A key problem is to limit the exchange of private preference information among the process stakeholders.

In this paper we propose a multiagent system made up of a set of autonomous, behaviour-based agents that act on behalf of their beliefs, preferences and objectives. All agents aim to reach a common classification but have private preferences.

The group classification is derived through a negotiation process in a cooperative context. A mediator agent is charged with the task of conducting the process. For each item, classifications are proposed to all the involved user agents in order to collect their feedback.

Our system has been developed with two objectives : the first is limiting the exchange of private information, while the second is to provide solutions that are fair to agents. Current ongoing works consist in evaluating our method, with respect to these two objectives.

Keywords

Multi-agent systems, ordinal classification, multilateral negotiation, group-based decision making, simulations.

1 Introduction

Collective decision making intervenes in situations where a group of decision makers, with individual preferences, have to make a common decision. For instance, accepting an amendment in a parliament is a group decision making problem. Moreover, in internet applications, groups of people need to agree on a joint decision in a variety of situations, as for example setting up meetings, planning vacations, watching movies together, to name but a few.

Several Group Recommender Systems (GRS) methods [2, 3] have been developed in recent years in order to deal with group decision making problems. Most of these methods, derive common decisions based on either aggregation procedures [1, 4] or negotiation techniques [6, 8]. In these latter works, agents interact by following different

protocols and strategies related to the agents' nature such as their ability to make concessions or to share information, until they reach an agreement.

An ordinal classification problem [5, 7] consists in partitioning a set of items into predefined ordered equivalence classes, called categories.

In this paper, we propose a multiagent decision making process aiming at deriving common classification from the preferences of several involved agents. This is done through a negotiation process in a cooperative context. To take into account different behaviours of multiple users, we use multiagent systems (MAS) [9], representing a group of autonomous agents, capable of reasoning, planning and communicating with each other.

In this work, we use a mediator agent to interact with the different user agents. The former proposes a classification for a given item, and, based on the agents feedback, decides whether to adopt the classification. The agents' behaviors are dynamic, so that agents can be increasing tolerant to accept classifications that do not completely match with their local preferences, in order to help the group to converge to a common decision. Communication between the agents is performed using a simplified agent communication language.

The protocol is developed with the goal of limiting the exchange of private information, hence, agents do not reveal entirely their preferences. We also conjecture that such protocol may produce solutions that are fairer (in the sense that, roughly speaking, the least satisfied agents do reasonably well) than a centralized approach.

2 Overview of the Proposed Method

Problem setting This work is concerned with an ordinal classification problem involving several agents (each acting on behalf of a human user). The problem consists in assigning a set of items \mathcal{X} to one of the categories in $\mathcal{C} = \{C_1, \dots, C_q\}$ with $q \geq 2$. The categories in \mathcal{C} are ordered; C_q is the best category and C_1 is the worst (the order is the same for all users).

We assume that each user is represented by a user agent; we denote the set of users agents as $\mathcal{N} = \{Ag_1, \dots, Ag_p\}$

with $p \geq 2$. We also assume a mediator agent, denoted by Ag_{Me} . We consider that each agent has a preferred category for each items in \mathcal{X} ; moreover a user agent prefers assigning an item x as close as possible (with respect to the order of \mathcal{C}) to his preferred category.

Our solution consists in a procedure where the mediator agent repeatedly proposes a given classification to all user agents. The mediator collects the feedback from the user agents and makes a classification when the number of acceptances surpasses a given threshold. Each agent then decide whether to accept a proposition or not.

The protocol The mediator maintains a set of items that have not been classified yet (initially all items). The algorithm proceeds by assessing, at each round, a different item x . The mediator agent (Ag_{Me}) proposes to the user agents the assignment of x to each of the categories in \mathcal{C} . It then waits the agents' replies; afterward it assesses the set of categories for which there exists a sufficient majority of agents accepting the proposed classification. In case this set is empty he postpones the assignment of x (placed at the end of the queue of the items to evaluate). Otherwise, x will be assigned to the category accepted by the highest number of agents (breaking ties in an arbitrary way).

The protocol continues until all items have been classified. As discussed below, the agents may become less or more tolerant in the subsequent rounds.

Acceptance strategy At each round, each user agent has to provide feedback (acceptance or rejection of the proposed classification) to the mediator agent. When the proposal fully satisfies the agent (the mediator proposes to assign an item x to the most preferred category of the agent), the agent accepts the proposal. Moreover, the agent can accept a proposal that does not fully satisfy him. To model an agent's disposition towards accepting a proposal, each agent is represented by an agent profile that represents the tolerance of agent with respect to the number of acceptable deviations (maximum displacement from the preferred class), the agent's flexibility regarding the number of concessions made all along the process and the agent's flexibility with respect to the negotiation time. Each user agent $ag_i \in \mathcal{N}$ is associated to three sub-utility functions, u_i , fl_i , and f_i , representing, respectively, the tolerance from accepting a classification other than the agent's preferred one, the preference for achieving a decision sooner rather than later, and the disposition to make new "concessions".

The aggregated utility $U_i(x, C_k) : \mathcal{X} \times \mathcal{C} \rightarrow [0, 1]$ combines u_i , fl_i and f_i ; its formula is agent-specific. For example, for the agent Ag_i , we could have :

$$U_i(x, C_k) = u_i(d_l) \times \frac{(fl_i(t_j) + f_i(nc_h))}{2}$$

where :

- d_l , with $l \in [0, q - 1]$, is the the number of categories (deviation) between the agent's classification of x and the proposed classification.
- t_j : The negotiation timestep discretized in m rounds, where $j \in [1, m]$. For instance, with the

passing of time the agent is more inclined to accept the proposed classification.

- nc_h , with $h \in [0, m]$, is the number of concessions (the number of times the agent accepted a proposal that does not fully satisfy him) made by the agent in each of the m rounds. For instance, the more concessions the agent Ag_i has made in the past, the less he is inclined to accept the classification.

Agent Ag_i accepts or rejects the proposal concerning the assignment of x to category C_k by comparing the value $U_i(x, C_k)$ with respect to an acceptance threshold, that we denote by λ (which is fixed for all agents along the process). In other words, the agent replies "accept" if $U_i(x, C_k) \geq \lambda$ and replies "reject" otherwise.

Current work We have implemented the proposed approach using the platform JADE and we are currently conducting simulations to assess the performance of the method with respect to several criteria (time required to find an agreement, average users satisfaction and fairness of the final solution, amount of disclosed private information) comparing it to a centralized approach.

Références

- [1] Linas Baltrunas, Tadas Makcinskas, and Francesco Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 119–126. ACM, 2010.
- [2] Punam Bedi, Sumit Kumar Agarwal, Vinita Jindal, et al. Marst : multi-agent recommender system for e-tourism using reputation based collaborative filtering. In *International Workshop on Databases in Networked Information Systems*, pages 189–201. Springer, 2014.
- [3] Iván Cantador and Pablo Castells. Group recommender systems : new perspectives in the social web. In *Recommender systems for the social web*, pages 139–157. Springer, 2012.
- [4] Inma Garcia, Sergio Pajares, Laura Sebastia, and Eva Onaindia. Preference elicitation techniques for group recommender systems. *Information Sciences*, 189 :155–175, 2012.
- [5] Vincent Mousseau, Roman Slowinski, and Piotr Zielniewicz. A user-oriented implementation of the ELECTRE-TRI method integrating preference elicitation support. *Computers & OR*, 27(7-8) :757–777, 2000.
- [6] Rachel Perez Palha. Negotiation throughout flexible and interactive tradeoffs applied to construction procurement. *Automation in Construction*, 99 :39–51, 2019.
- [7] Bernard Roy. The outranking approach and the foundations of electre methods. In *Readings in multiple criteria decision aid*, pages 155–183. Springer, 1990.
- [8] Christian Villavicencio, Silvia Schiaffino, J Andres Diaz-Pace, and Ariel Monteserin. Group recommender systems : A multi-agent solution. *Knowledge-Based Systems*, 164 :436–458, 2019.

- [9] M Woodridge and NR Jennings. Intelligent agents : Theory and practice. *The Knowledge Engineering Review*, 1995.

A multi-agent system approach to dynamic ridesharing for older people – State-of-the-art work and preliminary design

I. Lohja^{1,2}, Y. Demazeau³, C. Verdier¹

¹ Univ. Grenoble Alpes, LIG, 38000 Grenoble, France

² Mobicoop, 54000 Nancy, France

³ Univ. Grenoble Alpes, CNRS, LIG, 38000 Grenoble, France

iris.lohja@univ-grenoble-alpes.fr

Abstract

Ridesharing has disrupted the entire transport sector in the last decade, causing a cultural shift in the general population, but not so much in the older generations. In a previous paper, we proposed a multi-agent systems (MAS) approach for modelling and building an innovative ridesharing service tackling this lack of utilisation by older people. After having identified the obstacles that are impeding this utilisation, we dealt with each through MAS techniques. In this paper, we investigate the state of the art around these MAS techniques and describe the competing models for modelling each component of our multi-agent system, IMOPOP.

Keywords

ridesharing, multi-agent systems, agent-based modelling.

Résumé

Le covoiturage a perturbé le secteur des transports au cours de la dernière décennie, provoquant un changement culturel dans la population générale, mais pas tant dans les générations plus âgées. Dans un article précédent, nous avons proposé une approche multi-agents (SMA) pour la modélisation et la construction d'un service de covoiturage innovant pour remédier à ce manque d'utilisation par les personnes âgées. Après avoir identifié les obstacles qui entravent cette utilisation, nous les avons traités à travers des techniques SMA. Dans cet article, nous étudions l'état de l'art autour de ces techniques SMA et décrivons les modèles concurrents pour modéliser chaque composant de notre système multi-agents, IMOPOP.

Mots-clés

covoiturage, systèmes multi-agents, modélisation agents.

1 Introduction

Ridesharing has seen a lot of new advancements over the recent years, introduced by companies such as Uber in the international scene, and BlaBlaCar in France. Ridesharing is a service that offers one-way or two-way transportation to multiple participants on short notice. There are various reasons to use ridesharing : to reduce financial costs, carbon dioxide (CO_2) emissions and traffic congestion ; to

stay mobile ; to interact with other people ; all resulting in significant environmental and social benefits. Ridesharing has also provided a solution for the lack of car and public transport, and has increased the level of mobility independence. But unless there is willingness to participate and trust in ridesharing, a ridesharing service is not effective. We showed this to be true for the ridesharing service of French ridesharing company Mobicoop [33]. The company's data analysis revealed that older people¹ is the population *stratum* that uses its ridesharing service the least. Furthermore, researching the French ridesharing industry, we came to the conclusion that older people were not utilising² other ridesharing services either. Having said that, we define our research problem to be : how can we build a complete ridesharing service that older people will utilise ? To tackle this problem, we take on a multidisciplinary approach combining method engineering, artificial intelligence (AI), business models, public policies, and digital inclusion. In this paper, we address our research problem from the AI point of view, investigating the state of the art around the techniques that we consider in our work. We propose to construct a multi-agent system. Mobility services are inherently multi-agent, due to a large number of geographically distributed passengers, drivers, vehicles, and service providers. Moreover, a MAS approach allows for modularity, flexibility, autonomy, and scalability of the system.

We currently employ ridesharing as the only transport mode. What makes our context special is that older people do not generally have fixed daily activities, such as school or work, but rather sporadic and sometimes urgent activities. So, we are not dealing with conventional static ridesharing but with real-time or dynamic ridesharing³. Older people might also change their minds easily and often, especially if their confidence in the service, in the driver and in the co-riders is not ensured. Furthermore, other crite-

1. We consider older people to be people of 60+ years old.

2. Utilisation implies the action of making effective use of the service.

3. Dynamic ridesharing, unlike static ridesharing, includes single or recurring ridesharing trips with no fixed schedule, organised on a one-time basis, with a matching of participants taking place up until their departure time.

ria such as gender of and acquaintanceship with co-riders and smoking, which research have shown to be important constraints in ridesharing [31, 21], are amplified for older people.

The paper is organised as follows : next section gives a brief overview on the approaches for transport demand modelling and agent-based modelling. Section 3 presents our multi-agent model of the ridesharing system. Section 4 concludes the paper and discusses perspectives for future work.

2 Related Work

The study of transport systems has started with the analytical modelling methodology [24]. Analytic models are based on mathematical programming and continuous approximations., what makes these models too abstract for expressing some elements of various transport systems. To deal with the abstract nature of these models, simulation modelling was introduced [39]. Simulations are a powerful tool for analysing the behaviour of transport systems, but centralizing it, limits the interactions and negotiations among individual actors. The agent-based simulation paradigm [30] is used to overcome this shortcoming.

One important aspect of transport system modelling is the modelling of travel demand and behaviour. Travel behaviour research studies the physical movement of people outside their reference locations, i.e., the place a person returns to at the end of the day, for any activity. Two types of approaches to understanding travel behaviour that have prevailed over the years include trip-based approaches [35] and activity-based approaches [23]. The conventional trip-based approaches place trips as the fundamental unit of analysis and include four steps : trip generation, trip distribution, mode choice, and route choice. These approaches explicitly ignore the spatial and temporal constraints of travel behaviour, along with the fact that travel demand derives from the demand for activity participation. These weaknesses make trip-based approaches not adapted to our context. Activity-based approaches, on the other hand, consider decisions concerning the activities that trigger the demand for travel. These decisions include decisions on participation, place, time, duration, and co-riders. This type of approaches has begotten several conceptual frameworks and mathematical models, but it has failed to get the attention of empirical transport research.

Agent-based simulation and activity-based modelling are popular approaches in transport research [5]. Some relevant tools include MATsim [1] and STARCHILD [41]. However, because these simulations treat travellers as passive data structures updated synchronously at infrequent time intervals, they fail to take account of frequent within-the-day decisions and negotiations which are necessary for a dynamic system such as the ridesharing system. In order to cover for these drawbacks, the fully agent-based approach is used. The AgentPolis platform uses this approach for modelling multi-modal transport systems [26] and for implementing ridesharing models [19, 27]. In the former,

AgentPolis provides support for modelling the environment and agent-environment interactions, but the support for modelling agent behaviour is relatively basic; in the latter, machine learning methods are used to quantify the potential of ridesharing in reducing traffic load and total vehicle miles travelled.

Related work in ridesharing has been mainly on ride-matching problems [20, 25], inter-modality and multi-modality [38, 44, 46], risks associated with ridesharing [17, 32], and privacy issues [3, 49]. Others opted for Decision Support Systems (DSS). [13] proposed a DSS for the organisation of the transportation of cruise passengers in a touristic city, addressing multiple goals simultaneously : limiting traffic congestion, maximizing tourism penetration, limiting the impact on the city transportation capacity.

3 Multi-agent model of a ridesharing ecosystem

A ridesharing ecosystem is composed of humans interacting through a software application. Typically, a user in a ridesharing application could be both a driver and a passenger, even at the same time, but for different rides. This adds a dynamic organisation to our ecosystem. Additionally, a driver and a passenger could have different objectives when offering or requesting a ride. These include mainly the need for transport and lower costs for passengers, and lower costs, environmental issues and desire to help for drivers.

In this section, we take on a MAS approach and represent our ecosystem as a multi-agent system, which we call IMOPOP, where the agents represent the users of the ecosystem. In Subsection 3.1, we first describe shortly the obstacles that we have found to be blocking older people from using ridesharing services. Then, in Subsection 3.2, we investigate existing MAS techniques to deal with the identified obstacles, and MAS models to model each component of IMOPOP.

3.1 Analysis of the ridesharing ecosystem

Prior to the work presented here, we used a continual innovation method to analyse the ridesharing ecosystem with the objective of identifying, among others, the relevant actors and their functions [33]. Then, through interviews and focus groups, we diagnosed the ecosystem and found three major Blocking Points (BP) : trust, user preferences and accompaniment. The trust BP refers to older people's tendency to not trust strangers, which strongly affects their participation in ridesharing. The user preferences BP encompasses the very specific and well-determined preferences of older people, e.g., strong feelings against smoking in the car, preference of travelling with a female driver. The accompaniment BP deals with some older people's need for some level of assistance to e.g., get in and out of the car or to be accompanied to their place of origin and destination. Here, we deal only with the first two BPs, as addressing the accompaniment BP is not within the scope of this paper (this has been done in [33]). In the next sub-

section, we first present the methodology for building our multi-agent system, then we model each of its components.

3.2 IMOPOP – Architectural Design

3.2.1 Methodology

In [10], a population in a multi-agent system is described as consisting of the set of agents that inhabit it, along with the set of their possible behaviours and the set of interactions that they can have. We classify different types of actors in the ridesharing ecosystem into different agent populations in IMOPOP due to their different characteristics. As a result, we have three agent populations : drivers, passengers and intermediaries. Intermediaries include people, associations and organisations that act as an intermediary between the passenger (the older person) and the driver. The agents interact among them and with organisations in the MAS through coordination mechanisms. Coordination is defined as the management of dependencies between activities [34]. According to [48], these dependencies can be of various types, and the type of dependencies influences the choice of the coordination paradigm for the MAS. Most of the coordination paradigms are centred on one or some of the MAS components. In order to consider the dynamics at the level of each component, we use the Vowels paradigm [11], which offers a componential view of a MAS.

3.2.2 Modelling

One of the main elements of the ridesharing ecosystem is the Ridesharing Software (RS). Jennings and Wooldridge argue that taking on an agent-oriented approach is adapted to complex real-world software systems [28]. A well-known conventional software development paradigm is object-oriented programming, which found a specialisation in the agent-oriented paradigm coined by Shoham [42]. To identify the components of IMOPOP, we use the AEIO toolboxes approach to agent-oriented programming (AOP), which identifies four main components – Agents, Environments, Interactions, and Organisations [9]. Our purpose in this subsection is to find the most appropriate model for modelling each MAS component, starting with the agents.

Modelling agents

In literature, there are various definitions of an agent in a multi-agent system depending on the application. In this paper, we adopt the generic definition given in [14], which defines the agent as “An entity which is placed in an environment and senses different parameters that are used to make a decision based on the goal of the entity. The entity performs the necessary action on the environment based on this decision.” The main agent in IMOPOP is the older person. Thus, for modelling the agents, we focus on modelling older people. For this, we start our research in the social sciences.

One of the social science theories often used in socio-technical systems is Activity Theory (AT) [47]. AT has been developed in two schools of thought with the common objective of understanding human behaviour. In its origins in Russia, AT specifies that the behaviour of humans can be understood by examining the activities and the

actions constituting the activities wherein people engage. It has been used to understand individual human behaviour by examining the relationship between the Individual, the Object and the Mediating Artefacts (tools), whereby each component affects the other and the activity as a whole [47]. In the Scandinavian school of thought, AT was reformulated by Engeström [16] by adding an interacting entity to the original scheme of Individual, Object and Mediating Artefacts – the Community. Engeström’s scheme includes notions from human-computer interaction theory as well. It has been used for analysing individuals and groups, mainly to understand how technologies are adopted and adapted through use in complex social situations [22, 29]. The first AT school of thought inspired many works on modelling social behaviour, including Clancey *et al.*’s [8]. Here, they develop a cognitive model relating goals, knowledge, behaviours, and social concepts in the Brahms modelling and simulation system [7].

A Brahms model formalises observations in order to have a systematic way of relating different sources of information and use them in work system design. To get a Brahms model, participant observation is necessary. In our context, it would be fairly impossible to capture the many possible situations and the emerging diverse behaviours, which leads our work towards Situation Theory [2].

Situation Theory is a theoretical approach, according to which a system is defined as the composition of situation-related concerns. To collect information about these concerns and to address the user preferences BP, we use the Hodges’ Health Career Model (H2CM) [15]. It is a model centred around the person, based on the situation, considering all contexts (or as many as possible), and allows for simple modelling. One of the main reasons why this model is adapted to our context is that it allows for individualism and holism at the same time.

We use H2CM to structure knowledge or features of older people around two axes – the vertical axis of individual and group and the horizontal axis of humanistic and mechanistic – which create four knowledge domains – Interpersonal, Sciences, Sociology, and Political. The vertical axis indicates our primary focus on the older person as an individual and as a group. The horizontal axis separates features into humanistic – related to the social aspect of the older person/people – and mechanistic – related to the technical aspects concerning the older person/people. Figure 1 shows the H2CM model for the passenger population, where we can see some features identified and classified into the four knowledge domains.

Modelling the passenger population with H2CM in IMOPOP helps to identify and include in the system their preferences and constraints, thus addressing the user preferences BP. A UML representation of a passenger agent is shown in Figure 2.

Table 1 compares the approaches we have considered for the modelling of the agents, according to our main criteria.

Modelling the environment

The environment in a multi-agent system is the place where

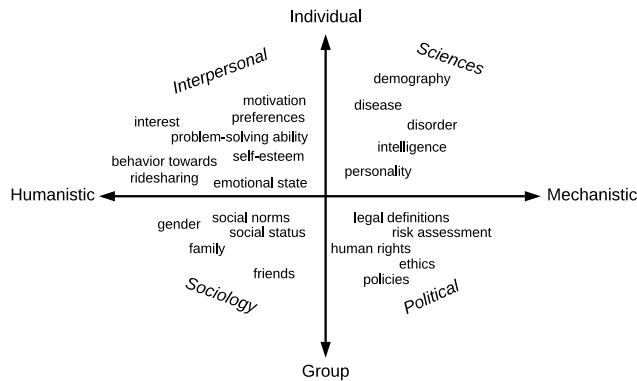


FIGURE 1 – Hodges’ Health Career Model for older people. The positioning of features within a quadrant is irrelevant.

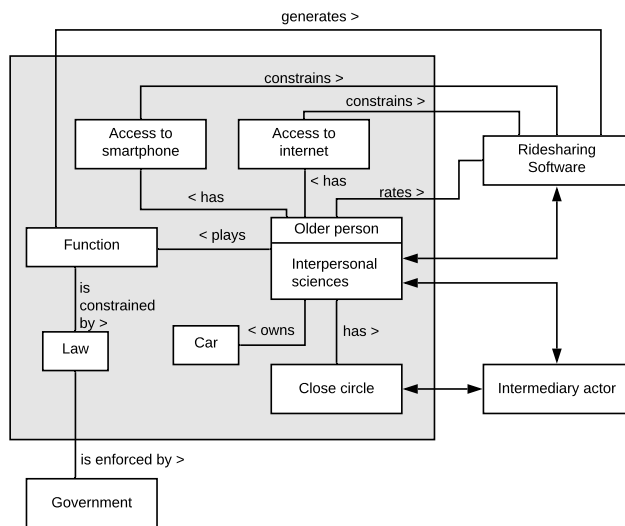


FIGURE 2 – UML representation of an older person agent in IMOPOP. The agent itself is shown in the grey square.

the agents are located [14]. An agent gathers information from the environment through its senses, which helps it in decision making. Different features of the environment affect the complexity of the MAS [48] – accessibility, determinism, dynamism, and continuity. The environment in IMOPOP is the physical world, where users interact through the RS. It is : inaccessible because agents sense incomplete and noisy data; non-deterministic because the outcome of agent actions cannot be entirely predictable; dynamic because the environment can change not only as a result of the actions of the agents; and continuous because the agents can enter any unlimited state. The different types of data that the agent can sense from the environment are referred to as parameters in our definition of an agent. These parameters concern Points Of Interest (POIs) for the agents, such as pharmacies, general practitioners, supermarkets, leisure time venues, and so on. They include

| | individual-focused | group-focused | situation-focused | user participation required |
|--------------------|--------------------|---------------|-------------------|-----------------------------|
| Activity Theory | | | | |
| - Russian SoT | ✓ | ✗ | ✗ | ✓ |
| - Scandinavian SoT | ✓ | ✓ | ✗ | ✓ |
| Situation Theory | | | ✓ | ✗ |
| H2CM | ✓ | ✓ | ✓ | |

TABLE 1 – Comparison between different approaches for the agent modelling on the basis of several criteria.

POIs’ position, opening and closing hours; the distance of the agent from the nearest public transportation; and other additional information. Furthermore, we include in the environment the ridesharing market represented by the main (competitor) players.

Modelling organisation

Demazeau and Rocha Costa [10] use the process-oriented notion of organisation, defining an organisation in a multi-agent system as a set of organisational roles and organisational links. An organisational role is the set of the agent’s behaviours which are part of the processes performed by a group of agents, while an organisational link is the mutual influence between agents in the group of agents performing a specific process in the system.

The agent organisation in IMOPOP is Mobicoop – the organisational roles of the agents in Mobicoop’s global processes and the organisational links among them.

Modelling interactions

Agents in a multi-agent system interact in order to achieve an ultimate goal – a shared ride, in the case of IMOPOP. During interactions, agents are in contact with each other directly (e.g., a driver and a passenger agree, in person, on sharing a ride), through another agent (e.g., a driver and a passenger are put into contact with each other through an intermediary), or through the environment (e.g., through Mobicoop’s RS). Agents can participate in different kinds of interactions [18]. According to goal compatibility, agents cooperate when their goals are compatible, or compete when their goals are incompatible. According to resource availability, agents are in conflict when resources are insufficient. According to agent ability, agents can collaborate when they have insufficient ability to reach their goals. Agents in IMOPOP have insufficient resources and skills to achieve their compatible goals. For this reason, they coordinate and collaborate with each other. We use an interaction model separately from the agent internal architecture. Beer *et al.* [4] discuss about the different forms that negotiation between agents can take – protocols, auctions and argumentation. Considering that our target group is older people, agents in IMOPOP should follow a very simple communication line, which should result in a mutually acceptable agreement. For this, the most appropriate form of negotiation are negotiation protocols, and the one that accomplishes this result with the minimum amount of communication is the Contract Net Protocol (CNP) [43].

CNP allocates tasks among autonomous agents in a typi-

cally spatially-distributed application, aiming to provide higher speed, reliability and extensibility in the application. The main problem in our application is the connection between agents with rides to be executed and the most appropriate agents to execute those rides. The contract net in our system is the set of agents in IMOPOP. Similarly to the user who can act as either a passenger or a driver, each agent can take on one of two roles, manager or contractor, but not both for a single interaction. For a particular interaction, one agent assumes the role of the manager – responsible for monitoring the execution of the ride – and another agent the role of the contractor – responsible for the actual execution of the ride. Typically, the manager announces the trips for which the passenger needs a ride. Then the contractor evaluates announcements for rides and bids on the ones suited to it. The manager receives the bids, evaluates them and awards a contract to the most pertinent contractor. The contract could be passed at the end of the first interaction, or the manager and the contractor can continue exchanging request and information messages, such as for adding a stop during the trip, once a contract has been awarded. The interaction is done by sending messages via the RS. These messages can be ride announcements, ride bids, ride contract awards, and others.

An example of an interaction between users in a ridesharing platform is given in Figure 3.

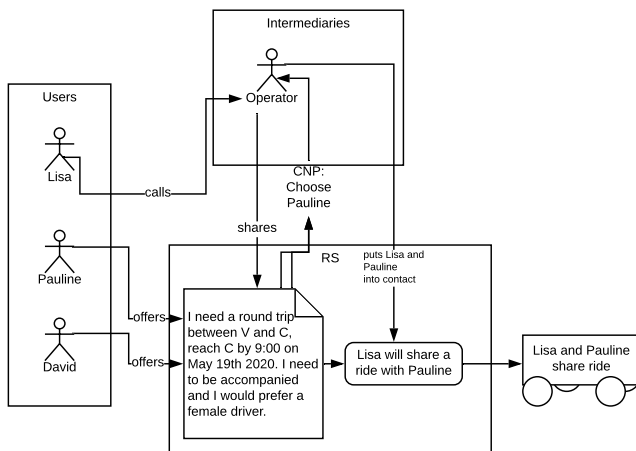


FIGURE 3 – Example of an interaction between users

Following, in Figure 4, we give the negotiation between the agents using CNP. Figure 4a shows a ride announcement message, where the Manager (for Operator) broadcasts its message for a ride to all the users of the RS, describing what Lisa needs (task abstraction), the criteria that a driver must meet in order to be able to submit a bid (eligibility specification), the expected form of a bid (bid specification), and the deadline for receiving bids from drivers (expiration time). We see in this message Lisa’s constraint and preference for the driver – willingness to accompany and female gender, respectively – which result from the modelling of Lisa as a passenger agent through H2CM. Figure 4b shows the bid message of the Contractor (for Pauline)

for the Manager’s announced ride. The same format follows the other contractor’s (representing David) bid message. These messages preserve the same bid format specified by the Manager in Figure 4a. The Manager, after considering both ride bids, determines that the one made by the Contractor (for Pauline) satisfies its constraint and preference, and awards a ride contract to it (Figure 4c). The Operator is notified about the contract award and subsequently puts Lisa and Pauline into contact.

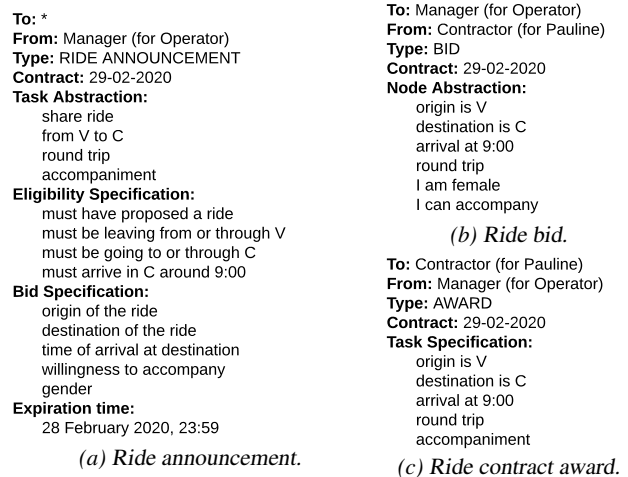


FIGURE 4 – Three messages in the example of the contract net of users negotiating for a ride through a ridesharing software

Modelling time

A passenger and a driver start negotiating for a ride at the time of negotiation. The negotiation can last for a short time or up until expiration time (as shown in Figure 4a). From the time of negotiation until the award of a ride contract, both passenger and driver could change their minds, availabilities, preferences, or constraints. For this reason, we add a clock in IMOPOP, which ensures that agents are intrinsically real-time. This concept comes from [42], where an agent is considered to have a mental state made of beliefs, capabilities, decisions, and obligations. At regular intervals, agents update their mental state based on previous changes (e.g., as a result of an interaction), and execute the commitments (e.g., a ride) for the current time. The clock makes sure that this two-step loop is initiated repeatedly at regular intervals. When a ride contract has been awarded, it means that at the time of execution, i.e., when the ride happens, the passenger, the driver, and either the driver’s or the passenger’s car are available⁴.

Modelling urgency and importance

There could be many various activities why older people (and people in general) need a ride. We categorize these activities into five main groups : health (e.g., for a medical appointment), family (e.g., to visit their child), leisure

4. We consider the case when the passenger has her/his own car but does not drive it, for various reasons. In such a case, the person acting as driver can drive the passenger’s car.

(e.g., to go to the theatre), shopping (e.g., for groceries), and sports (e.g., for a walk). Furthermore, they could determine a sequence of activities to be done during one single ride (tour). When there is only one passenger in a ride, the driver accommodates to the passenger's prioritised list of activities (in the case of a tour). Two more complicated cases include : when two or more passengers ask for different rides with the same driver at the same time, and when two or more passengers ask for the same ride with one driver. In these cases, we set priorities to the activities according to their urgency and importance [12]. These concepts are, however, subjective. As a result, the driver cannot satisfy everybody's priorities, so he chooses one out of the passengers in the first case, and the order of destinations (corresponding to the passengers' activities) in the second. Here comes into play the concept of trust.

Modelling trust

Trust is a subjective belief. Trust, in our context, concerns mainly the passenger and includes trust in the ridesharing service, in the driver and in the co-riders.

Work on trust in ridesharing is still limited : [50] designs a ridesharing system allowing people to gain information on their co-riders through social media, [37] embeds a ridesharing system within an organisation to incorporate a basic level of trust, [6] deals with drivers' trust in passengers, [36] tackles the trust in the platform.

In the ridesharing industry, we have found various elements employed by companies for building trust. Companies such as Citygo⁵ and Covoit'ici⁶ incorporate a rating system. Even though they are used in many human-computer applications, rating systems suffer from two main shortcomings : insufficient ratings at the system's start and ratings' biased nature. Some companies go a step further : Citygo⁷ rewards its users for giving their ratings, Keepmove⁸ provides trust by allowing users to create customised communities, and Smiile⁹ includes neighbours in the system while certifying that the ones on the platform are indeed neighbours in real life.

In MAS, passenger trust in the co-riders and the ridesharing platform are conceptualised as individual-level trust and system-level trust, respectively [40].

In IMOPop, we deal with the trust BP in four ways : by rating drivers and co-riders, a passenger can decide to either trust them or not based on their ratings ; by creating ridesharing communities with passengers and drivers, a passenger can (almost) completely trust members of her/his own community ; through the recommendation of her/his entourage, who is not necessarily a ridesharing participant, it becomes easier for a passenger to trust ; and through intermediaries, which are informed about the trustworthiness of ridesharing participants. Trust levels tend to differ in time. But depending on the means mentioned above, we can say that trust remains relatively static when built

through intermediaries (people will always trust intermediaries) ; trust built through communities is semi-dynamic ; trust built through ratings and people's recommendations is dynamic. Moreover, the dynamic evolution of the system brings along a possible change in the passenger trust in the ridesharing platform.

4 Conclusion and Future Work

The main objective in our work is to provide a ridesharing service that will be effectively used by older people. In another paper [33], we have shown that the older population hesitates to use ridesharing services and we have put forth three main blocking points for this utilisation – lack of trust in the service, driver and/or the co-riders ; very specific preferences which are sometimes difficult to be met ; and need for assistance. We have proposed to innovate in a ridesharing service by adopting MAS techniques. We have also proposed to model the ridesharing ecosystem as a multi-agent system, which we called IMOPop.

In this paper, we discussed about various MAS techniques for dealing with the first two blocking points. Viewing our multi-agent system as componential, we considered various models for modelling each of its components. We used the socio-technical Hodges' Health Career Model, which originates from the social care domain but we still described as it is crucial for IMOPop. It is used to model the older people as the agents in the system, addressing the user preferences blocking point. Given that we consider ridesharing as activity-based, we discussed about activities and proposed modelling the urgency and importance of these activities in the system. In the environment, we modelled agents' points of interest and the main players in the ridesharing market. Agents in IMOPop interact using the Contract Net Protocol. Time dynamics is modelled through a clock which makes agents intrinsically real-time. Finally, we addressed the trust blocking point by proposing four different means for building trust in IMOPop.

Future work includes the formalisation of IMOPop with a meta-model, the agent-based simulation using various scenarios and ultimately their validations. Afterwards, we will incorporate a business model into the multi-agent system and run other simulations.

Acknowledgment

This work is supported by the French Auvergne-Rhône Alpes region within the project MobiPA.

Références

- [1] K. W. Axhausen, A. Horni, K. Nagel, *The Multi-Agent Transport Simulation MATSim*, Ubiquity Press, 2016.
- [2] J. Barwise, J. Perry, *Situations and Attitudes*, MIT Press, 1983.
- [3] M. Baza, N. Lasla, M. Mahmoud, G. Srivastava, M. Abdallah, *B-Ride : Ride Sharing with Privacy-preservation, Trust and Fair Payment atop Public Blo-*

5. <https://www.citygo.me/>

6. <https://covoitici.fr/>

7. <https://www.citygo.me/>

8. <https://www.keepmove.fr/>

9. <https://www.smiile.com/>

- ckchain*, IEEE Transactions on Network Science and Engineering, pp. 1-1, 2019.
- [4] M. Beer, M. Luck, N. Jennings, Ch. Preist, M. Schroeder, *Negotiation in Multi-Agent Systems*, The Knowledge Engineering Review, Vol. 14, pp. 285-289, 1999.
- [5] B. Chen, H. H. Cheng, *A Review of the Applications of Agent Technology in Traffic and Transportation Systems*, IEEE Transactions on Intelligent Transportation Systems, Vol. 11, No.2, pp. 485-497, 2010.
- [6] X. Cheng, S. Linlin, B. Yang, *An investigation into sharing economy enabled ridesharing drivers' trust : A qualitative study*, Electronic Commerce Research and Applications, Vol. 40, 2020.
- [7] W. Clancey, P. Sachs, M. Sierhuis, R. Hoof, *Brahms : Simulating practice for work systems design*, International Journal of Human-Computer Studies, Vol. 49, pp. 831-865, 1998.
- [8] W. Clancey, M. Sierhuis, B. Damer, B. Brodsky, *Cognitive modeling of social behaviors*, Cognition and Multi-Agent Interaction : From Cognitive Modeling to Social Simulation, 2005.
- [9] Y. Demazeau, *FROM INTERACTIONS TO COLLECTIVE BEHAVIOUR IN AGENT-BASED SYSTEMS*, 1st. European Conference on Cognitive Science, pp. 117-132, 1995.
- [10] Y. Demazeau, A.C. Rocha Costa, *Populations and organizations in open multi-agent systems*, 1st National Symposium on Parallel and Distributed AI, 1996.
- [11] Y. Demazeau, *Steps towards multi-agent oriented programming*, 1st International Workshop on Multi-Agent Systems, 1997.
- [12] Y. Demazeau, D. Melaye, M-H. Verrons *A Decentralized Calendar System Featuring Sharing, Trusting and Negotiating*, Proceedings of the 19th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2006), pp. 731-740, 2006.
- [13] C. Di Napoli, P. M. Santamaria, S. Rossi *An Agent-Based DSS Supporting the Logistics of Cruise Passengers Arrivals*, Advances in Practical Applications of Scalable Multi-agent Systems. The PAAMS Collection, pp. 60-71, 2016.
- [14] A. Dorri, S. Kanhere, R. Jurdak, *Multi-Agent Systems : A survey*, IEEE Access, 2018.
- [15] M. Doyle, P. Jones, *Hodges' Health Career Model and its role and potential application in forensic mental health nursing*, Journal of psychiatric and mental health nursing, Vol. 20, 2012.
- [16] Y. Engeström, *Learning by expanding : an activity-theoretical approach to developmental research*, Hki : Orienta-konsultit, 1987.
- [17] M. Feeney, *Is Ridesharing Safe ?*, Policy Analysis, Cato Institute, No. 767, 2015.
- [18] J. Ferber, *Multi-agent systems - an introduction to distributed artificial intelligence*, 1999.
- [19] D. Fiedler, M. Čertický, J. Alonso-Mora, M. Čáp, *The Impact of Ridesharing in Mobility-on-Demand Systems : Simulation Case Study in Prague*, 21st International Conference on Intelligent Transportation Systems, pp. 1173-1178, 2018.
- [20] M. Friedrich, M. Hartl, C. Magg, *A modeling approach for matching ridesharing trips within macroscopic travel demand models*, Transportation, Vol. 45, pp. 1639-1653, 2018.
- [21] K. Ghoseiri, A. Haghani, M. Hamed, *Real-time Rideshare Matching Problem*, 2010.
- [22] J. Hardman, *Making sense of the meaning maker : tracking the Object of activity in a computer-based mathematics lesson using activity theory*, International Journal of Education and Development, Vol. 3, No. 4, pp. 110-130, 2007.
- [23] C. Heidemann, *Regional planning methodology : The first and only annotated picture primer on regional planning*, Institut für Regionalwissenschaft, Diskussionspapier No. 16, 1987.
- [24] D. A. Hensher, K. J. Button *Handbook of Transport Modelling*, Elsevier, Vol. 1, 2000.
- [25] L. Hou, D. Li, D. Zhang, *Ride-matching and Routing Optimisation : Models and a Large Neighbourhood Search Heuristic*, Transportation Research Part E Logistics and Transportation Review, Vol. 118, pp. 143-162, 2018.
- [26] M. Jakob, Z. Moler, A. Komenda, Z. Yin, A. X. Jiang, M. P. Johnson, M. Pěchouček, M. Tambe, *AgentPolis : towards a platform for fully agent-based modeling of multi-modal transportation*, Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, Vol.3, pp. 1501-1502, 2012.
- [27] M. Jakob, Z. Moler, *Modular framework for simulation modelling of interaction-rich transport systems*, 16th International IEEE Conference on Intelligent Transportation Systems, pp. 2152-2159, 2013.
- [28] N.R. Jennings, M. Wooldridge, *Agent-Oriented Software Engineering*, Artificial Intelligence, Vol. 117, pp. 277-296, 2000.
- [29] D. Jonassen, S. Land, *Theoretical Foundations of Learning environments*, Routledge 2nd edition, 2012.
- [30] F. Klügl, A. Bazzan, *Agent-Based Modeling and Simulation*, AI Magazine, Vol. 33, pp. 29-40, 2012.
- [31] I.P. Levin, M. K. Mosell, C. M. Lamka, B. E. Savage, M. J. Gray, *MEASUREMENT OF PSYCHOLOGICAL FACTORS AND THEIR ROLE IN TRAVEL BEHAVIOR*, 1977.
- [32] Y. Li, A. Taeihagh, M. De Jong, *The Governance of Risks in Ridesharing : A Revelatory Case from Singapore*, Energies, MDPI, Open Access Journal, Vol. 11, No. 5, pp. 1-21, 2018.

- [33] I. Lohja, C. Verdier, A. Front, *Towards a Utilized Ridesharing Service for Older People : A MAS Approach to Innovation*, HAL working paper or preprint :hal-02503704, 2020.
- [34] T. Malone, K. Crowston, *The Interdisciplinary Study of Coordination, Massachusetts Institute of Technology (MIT), Sloan School of Management, Working papers*, Vol. 26, 1993.
- [35] M.L. Manheim, *Fundamentals of Transportation Systems Analysis*, MIT Press, 1979.
- [36] X. Meng, X. Cheng, S. Fu, J. Sun, *Exploring Trust in Online Ride-sharing Platform in China : A Perspective of Time and Location*, Hawaii International Conference on System Sciences, 2020.
- [37] J. Morse, J. Palay, Y. Luon, S. Nainwal, *CarLoop : leveraging common ground to develop long-term car-pools*, Extended Abstracts Proceedings of the 2007 Conference on Human Factors in Computing Systems, 2007.
- [38] A. Najmi, T. H. Rashidi, W. Liu, *Ridesharing in the era of Mobility as a Service (MaaS) : An Activity-based Approach with Multimodality and Intermodality*, arXiv preprint arXiv :2002.11712, 2020.
- [39] T. I. Ören, B. P. Zeigler, *Concepts for Advanced Simulation Methodologies*, SIMULATION : Transactions of The Society for Modeling and Simulation International, Vol. 32, pp. 69-82, 1979.
- [40] S.D. Ramchurn, D. Huynh, N. R. Jennings, *Trust in multi-agent systems*, The Knowledge Engineering Review, Vol. 19, No. 1, pp. 1-25, 2004.
- [41] W. W. Recker, M. G. McNally, G. S. Root, *A Model of Complex Travel Behavior : Part I - Theory; Part II-Operational Model*, Transportation Research A, Vol. 20, pp. 319-330, 1986.
- [42] Y. Shoham, *Agent-oriented programming*, Artificial Intelligence, Vol. 60, pp. 51-92, 1993.
- [43] R.G. Smith, *The Contract Net Protocol : High-Level Communication and Control in a Distributed Problem Solver*, IEEE Transactions on Computers, Vol. C-29, No. 12, pp. 1104-1113, 1980.
- [44] M. Stiglic, N. Agatz, M. Savelsbergh, M. Gradisar, *Enhancing urban mobility : Integrating ride-sharing and public transit*, Computers & Operations Research, Vol. 90, pp. 12-21, 2018.
- [45] P. Tošić, R. Vilalta, *A unified framework for reinforcement learning, co-learning and meta-learning how to coordinate in collaborative multi-agent systems*, Procedia Computer Science 1.1, pp. 2217-2226, 2010.
- [46] S. Varone, K. Aissat, *Multi-modal transportation with public transport and ride-sharing - multi-modal transportation using a path-based method*, Proceedings of the 17th International Conference on Enterprise Information Systems, Vol. 1, pp. 479-486, 2015.
- [47] L.S. Vygotsky, *Mind in society : The development of higher psychological processes*, Cambridge, Mass. : Harvard University Press., 1978.
- [48] M. Wooldridge, *An Introduction to MultiAgent Systems*, Wiley Publishing, 2nd edition, 2009.
- [49] Y. Yuan, F. Wang, *Towards blockchain-based intelligent transportation systems*, 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 2663-2668, 2016.
- [50] Y. Zhou, Y. Huang, J. McGlynn, A. Han, *Who Will You Share a Ride With : Factors that Influence Trust of Potential Rideshare Partners*, 2017.

Approche décentralisée d'insertion avec amélioration continue de la qualité de la solution pour un système TAD

Alaa Daoud^{1,2}, Flavien Balbo^{1,2}, Paolo Gianessi¹, Gauthier Picard^{1,2}

¹ Univ Lyon, Mines Saint-Étienne,

² CNRS, Laboratoire Hubert Curien, UMR 5516, F-42023 Saint-Étienne, France.

alaa.daoud@emse.fr flavien.balbo@emse.fr paolo.gianessi@emse.fr gauthier.picard@emse.fr

Résumé

Dans le cadre du transport à la demande (TAD), l'appariement centralisé traditionnel permet d'obtenir des solutions optimales, mais reste NP-difficile et donc inadapté aux problèmes en ligne et dynamiques. Les approches décentralisées permettent d'obtenir des solutions réalisables en ligne, mais sans garantie de qualité. Ici, nous considérons une version dynamique du TAD, appelée Dial-A-Ride-Problem (DARP) où un réseau inter-véhiculaire est le support aux communications et les décisions sont prises de manière décentralisée par les véhicules. Dans ce cadre multi-agent, nous présentons un nouvel algorithme heuristique décentralisé d'insertion fondé sur une approche classique consistant à associer les véhicules aux demandes les plus proches dans le temps et l'espace, étendue par une phase d'optimisation afin d'améliorer la qualité de la solution. Nous évaluons ses performances sur des données synthétiques et la comparons à une approche gloutonne.

Mots-clés

TAD, DARP, V2V, décentralisation, SMA, heuristique d'insertion

Abstract

In the context of On-Demand Transport (ODT), traditional centralized matching provides optimal solutions, but remains NP-Hard and therefore unsuitable for online and dynamic problems. Decentralized approaches provide feasible solutions in real time, but without any guarantee of quality. Here, we consider a dynamic version of ODT, called Dial-A-Ride-Problem (DARP) where a V2V network is the communication support and decentralized decisions are made by vehicles. In this multi-agent framework, we present a new decentralized insertion heuristic algorithm based on a classical approach consisting in associating vehicles to the closest requests in time and space, extended by an optimization phase in order to improve the quality of the solution. We evaluate its performance on synthetic data and compare it to a greedy approach.

Keywords

On-Demand Transport (ODT), DARP, V2V, decentralized, MAS, insertion heuristic.

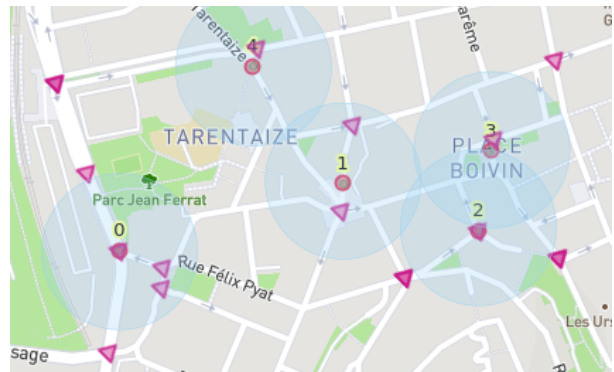


FIGURE 1 – Un exemple de problème AVFAP, avec les sources de demande (triangles) et les taxis (cercles) avec leurs portées de communication respectives (en bleu).

1 Introduction

Le concept de transport à la demande (TAD) a été formulé pour la première fois vers 1990 comme une solution à la désaffection croissante des utilisateurs potentiels des transports publics, en particulier la nuit. La recherche dans ce domaine se concentre principalement sur l'allocation des véhicules aux demandes, le routage et la gestion de flotte. L'affectation d'un ensemble de ressources à un ensemble d'agents est l'un des problèmes d'optimisation combinatoire les plus fondamentaux des plus étudiés dans la littérature [Danassis et al., 2019]. Il peut être résolu par programmation linéaire car sa relaxation de la formulation admet des solutions intégrales optimales à un instant donné t . Cependant, mise en oeuvre par un *dispatcher* central, cette approche, exige que les véhicules aient un accès continu à leur portail via l'infrastructure de communication globale (4G, par exemple), ce qui est coûteux et peut faire du portail un goulot d'étranglement critique. La complexité algorithmique du problème, qui étend le problème du voyageur de commerce (TSP), fait qu'il est difficile pour un *dispatcher* centralisé de gérer la dynamique du problème pendant son exécution (demandes en continue, problèmes de circulation et autres dynamiques de l'environnement). On peut s'attendre à ce que la décentralisation permette de faire face à ces problèmes.

L'heuristique d'insertion de Solomon [1987] est une méthode populaire pour résoudre une variété de problèmes d'ordonnancement et de routage. Elle peut être utilisée comme une méthode pour trouver rapidement une solution réalisable mais sans garantie de qualité de la solution. Dans le cas du problème de routage des véhicules (VRP), elle produit une solution en insérant de façon répétée des demandes non programmées dans un itinéraire partiellement construit ou comme première demande dans un nouvel itinéraire.

Dans ce travail, nous proposons le problème d'allocation de flotte de véhicules autonomes (AVFAP), qui étend le DARP traditionnel en considérant une flotte de véhicules autonomes qui se coordonnent sans *dispatcher* central. Les véhicules prennent des décisions décentralisées selon des informations échangées via une communication pair-à-pair (P2P). La communication véhicule-à-véhicule (V2V) via la communication dédiée à courte distance (DSRC) offre une faible latence, une connectivité réseau rapide, et une portée de communication allant jusqu'à 300 m [Dey et al., 2016]. Avec des portées de communication limitées (illustrée Figure 1), chaque véhicule n'est conscient que d'un sous-ensemble de demandes, et il ne communique avec une autre entité (un véhicule ou une source de demande) que si elle est directement à portée ou par transitivité. Nous proposons dans ce travail une nouvelle heuristique décentralisée pour l'AVFAP, bénéficiant de la forte réactivité des heuristiques d'insertion, et des bons résultats obtenus par l'optimisation par l'échange des demandes entre véhicules. Cet article est organisé comme suit. Dans la section 2, nous donnons un aperçu des efforts déployés dans la littérature pour résoudre ce problème. La section 3 expose notre définition du problème AVFAP. Ensuite, nous présentons notre solution pour assurer une prise de décision rapide par les agents dans la section 4 et notre proposition d'amélioration en continu est détaillée dans la section 5. Les paramètres des expérimentations, l'évaluation et les résultats sont détaillés dans la section 6. Enfin, nous concluons le document dans la section 7 avec quelques perspectives.

2 Travaux connexes

Les approches traditionnelles du TAD envisagent soit une architecture avec *dispatcher* centralisé comme dans [Egan et Jakob, 2015; Shen et Lopes, 2015] soit une architecture multi-agent (SMA) décentralisée pour réduire la complexité des problèmes comme dans [El Falou et al., 2014; Grau et al., 2013]. Dans ce travail, nous considérons cette seconde famille de travaux. Parmi les nombreuses approches pour les décisions décentralisées et l'auto-coordination, Glaschenko et al. [2009] introduit une solution de planification en temps réel fondée sur les offres de plusieurs agents dans des environnements entièrement décentralisés, dans laquelle chaque véhicule est représenté par un agent qui peut négocier par des canaux radio avec des critères de décision flexibles. Un algorithme de reconnaissance des formes est utilisé pour prédire les emplacements les plus probables pour la prochaine demande pour recommander les déplacements vers ces emplacements.

Étudiant l'applicabilité de la programmation génétique (GP) pour développer des SMA décentralisés qui résolvent le DARP, van Lon et al. [2012] présente une méthode pour générer automatiquement un SMA qui peut résoudre le DARP pour un ensemble spécifique de scénarios. La programmation génétique est utilisée pour générer une heuristique qui est efficace pour résoudre le DARP. Le meilleur résultat obtenu avec cette approche est la planification d'une seule demande à l'avance par véhicule, ce qui maximise les intérêts locaux de l'agent et produit une solution réalisable très rapidement.

Grau et Romeu [2015] ont proposé un modèle d'agent où l'appariement de l'offre et de la demande dépend des événements simulés du marché réel des taxis. Selon leur conclusion, l'une des principales limites du modèle est l'hypothèse d'une distribution uniforme de la demande dans la zone de service.

Pour remédier à l'incertitude causée par la nature dynamique des demandes en ligne, et en ignorant le temps nécessaire à l'exécution d'un algorithme de planification, Agatz et al. [2011] ont proposé d'utiliser une approche déterministe de solution à horizon glissant, dans laquelle les plannings sont établis en utilisant toutes les informations connues dans un horizon de planification, qui est "glissé" vers l'avant pour inclure plus d'informations. Dans nos travaux précédents [Picard et al., 2018], sur la base de la coordination des véhicules par communication V2V, nous avons proposé le modèle d'allocation de ressources en ligne avec contraintes de communication (OLC²RA). La solution proposée concerne une flotte de taxis autonomes, dans laquelle un véhicule décide de sa prochaine destination en ne considérant qu'une seule demande à l'avance. Au contraire, l'heuristique décentralisée ALMA de Danassis et al. [2019] est complètement découplée et ne nécessite pas de communication entre les participants. Ils démontrent une limite supérieure de la vitesse de convergence qui est polynomiale à la quantité désirée de ressources et d'agents concurrents par ressource ; Dans le cas réaliste où les quantités mentionnées sont limitées quel que soit le nombre total d'agents/ressources, le temps de convergence reste constant à mesure que la taille totale du problème augmente. Cependant, la détection des conflits nécessite toujours une communication avec d'autres véhicules, ressources ou une entité centrale pour permettre aux ressources de partager des informations sur leur statut, comme le mécanisme de coordination du tableau noir.

Jusqu'à présent et à notre connaissance, les efforts déployés dans le domaine de la planification DARP fournissent soit des solutions optimales avec un temps d'exécution souvent peu adapté à la dynamique, soit une solution solution réalisable (rapide est non nécessairement optimale) pour répondre aux demandes en ligne. Dans ce travail, comme la solution optimale pour ce problème dynamique est irréalisable avec un temps de calcul raisonnable, notre proposition combine les avantages de ces approches dans une heuristique qui fournit une réponse rapide fondée sur les enchères (solution rapide et faisable), puis cette solution est améliorée progressivement grâce à un protocole d'échange entre

agents des demandes dont la planification est à remettre en cause suite à l'évolution de l'environnement.

3 Modèle du problème AVFAP

Le problème de l'allocation d'une flotte de véhicules autonomes (AVFAP) étend le DARP traditionnel en considérant une flotte de véhicules autonomes qui se coordonnent sans *dispatcher* central. Les véhicules prennent des décisions décentralisées selon des informations échangées via une communication pair-à-pair (P2P). Le graphe de la carte de la ville, illustré dans la Figure 2, est constitué de nœuds qui représentent des emplacements géographiques et d'arcs qui représentent les liaisons routières entre ces emplacements. Une flotte de véhicules autonomes est répartie dans la ville, chaque véhicule possède un ensemble de propriétés dont les valeurs sont constantes (capacité, coût et vitesse moyenne) ou variables (emplacement, planning) car dépendantes du temps. Les passagers émettent des demandes à partir de différents endroits (que nous appellerons sources ensuite). Chacune prend la forme d'une requête qui définit : les lieux de prise en charge et de livraison associés à la fenêtre de temps de service souhaitée. Nous définissons une solution comme un planning pour chaque véhicule qui répond aux demandes en satisfaisant leurs contraintes, tout en minimisant le temps d'attente des voyageurs et le coût de déplacement des véhicules. Les véhicules communiquent par diffusion via un réseau V2V, où la portée de communication est limitée. Chaque véhicule qui reçoit une nouvelle information la rediffuse et les véhicules sont ainsi connectés par transitivité dans les limites des rayons de communication. Dans ce contexte, un véhicule ne connaît pas les demandes en dehors de sa zone de communication. Cette connaissance évolue pendant le temps d'exécution, car le véhicule se déplace, reçoit de nouvelles demande des sources de demandes, rencontre d'autres véhicules et échange des messages entre eux.

Ce modèle AVFAP est défini comme suit :

$$\begin{aligned} AVFAP &:= \langle M, V, D, T \rangle \\ M &:= \langle G, w \rangle \\ V &:= \{v_1, v_2, \dots, v_n\} \\ D &:= \{d_1, d_2, \dots, d_m\} \\ T &:= \{t_0, t_1, \dots, t_{end}\} \end{aligned}$$

Ici, M définit la carte des infrastructures urbaines (emplacements, routes et distances) sous la forme d'un graphique dirigé G associé à une fonction de valuation w ; l'offre est représentée par une flotte V de n véhicules autonomes; D définit un ensemble dynamique de demandes de passagers qui apparaissent au moment de l'exécution; et T définit l'horizon temporel dans lequel les véhicules doivent répondre aux demandes des passagers. Nous définissons le temps T comme un ensemble discret de *ticks*.

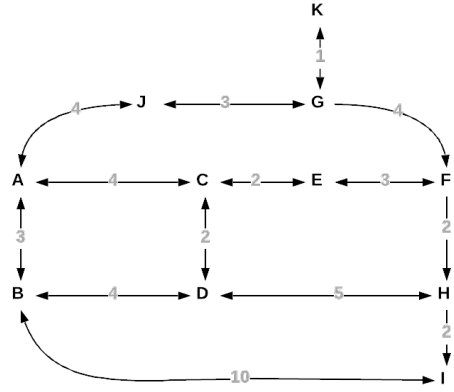


FIGURE 2 – Un exemple d'infrastructure urbaine

3.1 Propriétés de l'infrastructure routière

La carte des infrastructures routières est définie par un graphe dirigé pondéré M , comme le montre la Figure 2.

$$\begin{aligned} G &:= \langle N, E \rangle \\ w &:= \{w_{e_1}, w_{e_2}, \dots, w_{|E|}\} \end{aligned}$$

G est un graphe connecté dirigé où N est l'ensemble des nœuds, E est l'ensemble des arcs entre les nœuds. La fonction de valuation w associe chaque arc e à la valeur w_e selon une mesure de distance temporelle (par exemple, le temps de conduite moyen en minutes), qui sera utilisée pour calculer les coûts opérationnels des déplacements des véhicules.

3.2 Propriétés des véhicules

Chaque véhicule $v \in V$ est défini par

$$v := \langle capa, cpd, range \rangle$$

et ses propriétés dynamiques sont définies par

$$\begin{aligned} v_location &: V \times T \rightarrow N \cup E \\ free_seats &: V \times T \rightarrow \mathbb{N}^+ \end{aligned}$$

où $capa$ définit le nombre total de sièges passagers dans le véhicule, cpd définit le coût du véhicule par unité de distance, et $range$ définit la portée de communication dans laquelle le véhicule peut communiquer avec d'autres entités. A tout moment $t \in T$ un véhicule $v \in V$ connaît sa situation courante : $v_location$ définit l'emplacement de v qui pourrait être situé dans un nœud $n \in N$ ou se déplaçant sur un arc $e \in E$; et $free_seats$ définit le nombre de places libres disponibles dans v au moment t .

3.3 Propriétés des demandes

Une demande $d \in D$ est définie comme

$$d := \langle required, tw, pick_up, drop_off \rangle$$

Pour une demande $d \in D$, $required$ est le nombre de sièges requis; tw définit un intervalle de temps (t_{min}, t_{max}) dans

lequel l'événement de ramassage est considéré comme acceptable; *pick_up* et *drop_off* sont respectivement l'origine et la destination de la demande. Comme pour les véhicules, au temps $t \in T$, nous considérerons qu'une demande $d \in D$ peut également être communiquée en utilisant le réseau V2V. La demande pourrait être émise par le client ou l'infrastructure (unité de bord de route).

4 Insertion basée sur des enchères

Dans notre modèle, nous utilisons une heuristique d'insertion comme celle décrite par Solomon [1987] pour adapter en continu les plannings locaux des véhicules. Le résultat de cet algorithme est un ensemble de demandes avec pour chacune l'horaire auquel un véhicule sera à la position de son origine. Chaque agent détermine ses horaires pour maximiser la valeur de la qualité de sa solution. Comme plusieurs véhicules peuvent être intéressés par une même demande, nous avons besoin d'un mécanisme de coordination pour résoudre ces conflits. Nous utiliserons pour cela un mécanisme d'enchères, qui est l'un des moyens efficaces et éprouvés pour résoudre de tels problèmes [Cramton et al., 2007].

4.1 Critère de l'enchère

Lorsqu'un véhicule v a connaissance d'une demande d , il la classe dans sa file d'attente selon la priorité qu'il lui a attribuée.

Au temps t , v choisit la première demande d_s dans la file d'attente, génère un ensemble d'alternatives, chacune étant un planning potentiel résultant de l'insertion de d_s dans une étape réalisable du planning actuel de v . Le coût opérationnel marginal de l'ajout de cette demande au planning est noté *cost*. Le choix avec le meilleur *cost* est considéré pour diffuser une offre

$$Bid_v^d(t_{start}, cost)$$

avec t_{start} le moment de *pick_up* pour d_s .

Dans ce document, nous considérons le coût opérationnel d'un voyage comme la longueur totale de son trajet (somme des distances indiqués dans la Figure 2). Le coût marginal d'insertion est donc la différence de longueur de trajet entre

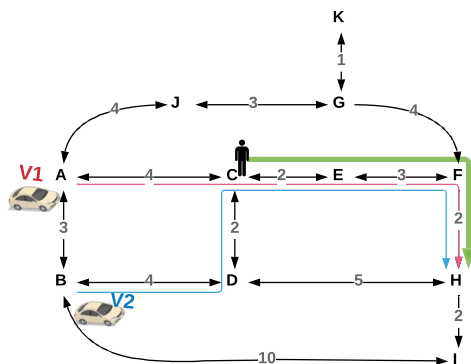


FIGURE 3 – V_1 remporte l'enchère pour servir d_1 de C à H

le trajet initial et le nouveau trajet. Les offres restent disponibles pendant une période de temps spécifique t_{expire} . Ainsi, si le coût de l'offre de v est inférieur à toute autre offre reçue à $t + t_{expire}$ pour servir une demande d , il se considère comme gagnant de l'enchère, et met à jour son planning avec le nouveau chemin de l'offre.

Exemple 1 Le scénario simple de la Figure 3 montre deux véhicules V_1 et V_2 situés en A et B, en état de repos avec des plannings vides au début. Au temps t_1 , la première demande est annoncée $d_1 : < 1, (t_{10}, t_{20}), C, H >$. Les deux véhicules connaissent maintenant d_1 . V_1 peut le servir en suivant le chemin $A \rightarrow C \rightarrow E \rightarrow F \rightarrow H$, donc V_1 place l'offre $Bid_{V_1}^{d_1}(t_{10}, 11)$. V_2 peut la servir via le chemin $B \rightarrow D \rightarrow C \rightarrow E \rightarrow F \rightarrow H$, donc émet l'offre $Bid_{V_2}^{d_1}(t_{10}, 13)$. V_1 est considéré comme un gagnant et ajoute d_1 à son planning, de sorte que le coût opérationnel global de la flotte est maintenant 11.

4.2 Le besoin d'échange de demandes

Afin de pouvoir faire des offres efficaces pour de nouvelles demandes ou pour améliorer la solution, nous proposons que les véhicules échangent leurs demandes planifiées.

Exemple 2 La Figure 4 montre une situation où l'utilisation de l'heuristique d'insertion fondée sur les enchères est très réactive mais ne garantit pas de bons plannings. Au moment t_2 où la nouvelle demande $d_2 : < 1, (t_{15}, t_{40}), J, K >$ arrive, les deux véhicules en prennent connaissance et placent leurs éventuelles offres. En l'absence de capacité d'échange, V_1 a encore d_1 dans son planning (avec un coût initial de 11), donc la meilleure offre qu'il peut placer est de servir les deux demandes (servir d_2 puis aller servir d_1) avec un coût marginal de 14. Alors que V_2 place l'offre gagnante $Bid_{V_2}^{d_2}(t_{15}, +11)$, il ajoute d_2 à son planning, et le coût global devient 22. Notez que dans ce cas, servir d_2 avec V_1 et laisser V_2 s'occuper de d_1 entraîne un gain de coût opérationnel global de 21, mais cette solution n'est jamais réalisée car d_1 est déjà programmé sur V_1 .

Pour parvenir à ce type de solution, nous devons permettre

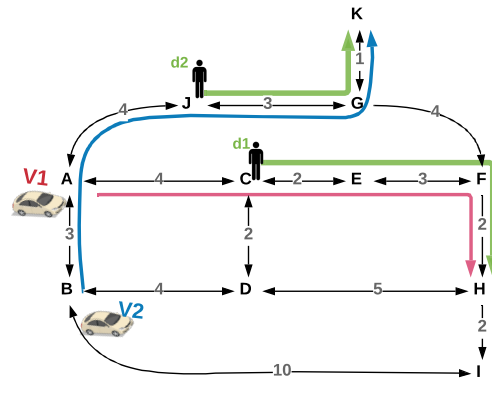


FIGURE 4 – Sans échange de la demande, V_2 gagne d_2 et V_1 garde d_1

aux véhicules d'échanger leurs demandes déjà programmées.

5 Améliorer la qualité des solutions

Dans ce qui suit nous proposons un protocole d'optimisation locale pour améliorer la qualité de la solution pour les cas comme celui illustré Figure 4 et l'exemple 3

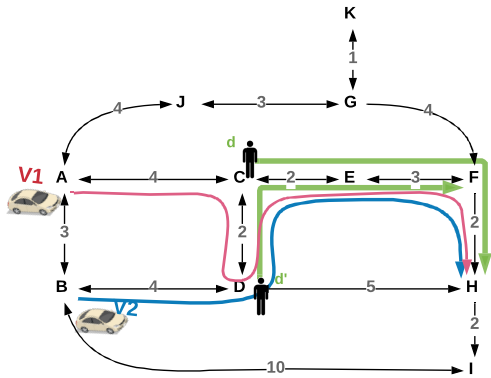


FIGURE 5 – Même s'il peut servir les deux demandes, V_2 ne peut soumettre une offre que pour une seule demande à la fois

5.1 Protocole d'optimisation « Pull-demand »

À l'instar de la stratégie *Rolling Horizon* de Agatz et al. [2011], nous proposons un protocole d'optimisation d'offres pour améliorer notre heuristique. Dans la stratégie *Rolling Horizon*, tous les plannings des véhicules sont considérés comme temporaires et disponibles pour être planifiés par n'importe quel véhicule, à moins qu'ils ne soient considérés comme des *demandes engagés* par des événements particuliers (par exemple, v a commencé à servir (se diriger vers) d , dont le temps restant pour le servir est inférieur au seuil de l'horizon). L'application de cette stratégie exige que tous les plannings des véhicules soient en mémoire partagée, de sorte que lorsqu'un véhicule v_i offre de servir une demande d , il sache s'il est programmé par un autre véhicule v_j , et donc s'il doit envoyer son coût d'offre à v_j . Ensuite, v_j calculera le gain (ou la perte) de coût opérationnel en abandonnant d et le coût pour v_i . S'il y a un gain, il accepte d'abandonner d et ensuite v_i met à jour son planning avec d , sinon l'offre est rejetée.

Dans notre proposition de protocole, nous ne faisons pas appel au concept de *demande engagée*, mais un véhicule ne peut faire des offres que pour des demandes qu'il peut satisfaire, de sorte que les demandes qui sont reprises ou qui n'ont pas assez de temps pour être reprogrammées sont automatiquement ignorées par l'agent. Une autre différence ici est que nous n'avons pas de mémoire partagée. Les agents échangent des informations sur le contexte de l'environnement et sur les demandes par des messages d'information. En plus de la proposition de Agatz et al. [2011], où l'optimisation est effectuée périodiquement à une fréquence prédéfinie, le protocole que nous proposons doit être exé-

Protocole 1 Protocole d'optimisation « Pull-demand »

Étape 1 Des nouvelles demandes entre dans le système par ordre d'annonce.

Étape 2 Chaque une est diffusée dans l'ensemble connecté auquel les sources appartiennent. Chaque agent de cet ensemble sélectionne ses demandes potentielles parmi toutes leurs demandes : les nouvelles demandes, les demandes planifiées et non planifiées qui n'ont pas encore atteint leur heure de départ prévue.

Étape 3 Les agents débutent l'enchère pour servir leurs demandes potentielles.

Étape 4 Chaque agent recherche parmi ses demandes planifiées celle à satisfaire au prochain tick, cette demande est appelée d_{next} . Si d_{next} existe, l'agent diffuse un message "clear_demand" pour informer les autres agents de sa prise en charge de d_{next} . Chaque récepteur efface cette dernière de leurs ensembles de demandes potentielles et connues. Chaque agent efface toute autre demande qui atteint sa limite de temps.

Étape 5 Les demandes programmées et non programmées qui ont encore du temps restent diffusées par leurs sources (*Étape 2*). Cela permet une meilleure planification dans le prochain tick, étant donné que de nouvelles demandes peuvent être annoncées.

cuté en parallèle avec une stratégie fondée sur des enchères d'insertion, sur la base des informations partagées sur le contexte courant pour avoir une replanification rapide pour les demandes en continu. Le **Protocole 1** détaille cette stratégie.

5.2 Discussion

Compte tenu du contexte décentralisé, l'utilisation de l'heuristique d'insertion est très efficace en termes de temps de réponse. La complexité temporelle de l'heuristique d'insertion de base pour le VRP est en $\mathcal{O}(n^3)$ [Campbell et Savelsbergh, 2004]. Ce type d'heuristique est souvent utilisé pour résoudre des DARP, où les nouvelles demandes entrantes doivent être traitées en continu et intégrées dans les plannings évolutifs des véhicules. Dans l'exemple suivant, nous illustrons comment le protocole « Pull-demand » peut apporter une amélioration notable de la qualité de la solution.

Exemple 3 Examinons un autre cas illustré dans la Figure 5, où V_1 possède d dans son planning, V_2 a un planning vide, et une nouvelle demande $d' : < 1, (t_{10}, t_{20}), D, F >$ est annoncée. Les offres des véhicules sont donc $Bid_{V_1}^d(t_{12}, +4)$, et $Bid_{V_2}^{d'}(t_{10}, +11)$. Examinons le protocole mis en œuvre, étape par étape.

Étape 1 d' entre dans le système à t_2 et les deux véhicules

en sont conscients, V_1 remporte l'enchère avec un coût global de 15

Étape 2 d et d' sont maintenant dans l'ensemble de demandes connues par les deux véhicules, V_2 calcule les coûts pour servir d seule (13), d' seule (9) et les deux demandes ensemble ce qui fait 13. Il sélectionne donc les deux demandes comme ses demandes potentielles. V_1 n'a pas de demande potentielle car il a déjà les deux demandes dans son planning

Étape 3 V_2 place une offre $Pull_Bid_{V_2}^{d',d}((t_{10}, t_{12}), 13)$. Pour V_1 le coût pour servir les deux demandes est de 15 donc il accepte $Pull_Bid_{V_2}^{d',d}$ car cela engendre un gain de 2.

Étape 4 Aucune des demandes n'atteint son temps de service prévu ou la limite supérieure de sa fenêtre temporelle.

Étape 5 Toutes les demandes connues restent diffusées et disponibles pour la prochaine amélioration potentielle.

V_2 gagne et la solution est améliorée avec un cycle d'optimisation supplémentaire.

6 Évaluation expérimentale

Dans cette section, nous évaluons expérimentalement la performance de notre approche « Pull-demand », en utilisant des données générées et des données cartographiques ouvertes.

6.1 Configuration expérimentale

La ville de Saint-Étienne a été choisie pour la simulation. La structure du graphique $G = \langle N, E \rangle$ incluant les nœuds, les arêtes et un ensemble de sources d'émission de la demande $S \subset N$ est extraite d'OpenStreetMap (OSM). Dans toutes les expérimentations, nous avons fixé le nombre de sources $|S| = 20$, ayant un ensemble $E_S \subset E$, tel que $|E_S| = 75$ arcs reliant les sources, chaque arc a un nombre de points qui varie en fonction de sa longueur et des informations extraites d'OSM. La distance entre deux points consécutifs est de 40 mètres. Nous avons utilisé un simulateur de transport en temps discret disponible dans la *Plateforme Territoire* pour évaluer la stratégie proposée et l'analyser en termes de qualité de service et de gain.

Une flotte V de n véhicules est distribuée au hasard sur S au début de l'exécution. Chaque véhicule $v \in V$ se déplace d'un point à un autre sur le même arc au cours de chaque cycle de simulation. Dans notre test, nous considérons que les véhicules communiquent via V2V avec une portée de communication réaliste de 250 mètres, de sorte qu'un véhicule peut envoyer/recevoir des messages vers/depuis d'autres entités situées dans sa portée. Chaque véhicule

v peut adopter deux comportements de déplacement distincts : soit *marauder* pour les demandes, soit *aller* vers une destination :

- *marauding* définit l'état d'un véhicule lorsqu'il n'a pas de demande à servir, c'est-à-dire qu'au début de la simulation, chaque véhicule maraude jusqu'à ce qu'il décide de servir une demande. Une fois qu'un passager est déposé, le véhicule repasse en *marauding*. Dans cet état, le véhicule se déplace aléatoirement dans son quartier afin de trouver des demandes à servir.
- *going_to* définit l'état d'un véhicule lorsqu'il a une destination spécifique, c'est-à-dire qu'une demande de desserte du véhicule est soit *going_to* lieu de prise en charge si la demande n'est pas encore prise en charge, soit *going_to* lieu de livraison sinon.

À chaque cycle de simulation, 0 ou 1 demande est générée de façon aléatoire uniforme. Pour chaque demande, les points d'origine et de destination sont générés de manière aléatoire et uniforme à partir de l'ensemble des sources. La fenêtre de temps pour les demandes est générée en utilisant deux paramètres constants l et u pour les limites inférieure et supérieure comme suit. $[tw_{min}, tw_{max}]$ est initialisée avec deux valeurs aléatoires uniformes où :

$$tw_{min} < tw_{max}$$

$$tw_{min} \geq t_{actuel} + l$$

$$tw_{max} \leq tw_{min} + u$$

Les critères d'évaluation de ces simulations sont principalement le nombre de demandes satisfaites comme mesure de la qualité de service (QoS), le bénéfice simulé de la solution comme mesure de la qualité des affaires (QoB). Le bénéfice est calculé en termes de différence entre le prix du voyage et le coût.

$$profit = total_income - total_cost$$

où

$$total_income = \sum_{d \in D_s} P + p * distance(d)$$

$D_s \subseteq D$ est l'ensemble de toutes les demandes satisfaites, P est un prix fixe par demande, p est un facteur de tarification par unité de distance parcourue, $distance(d)$ est la distance totale de déplacement pour une demande d et

$$total_cost = \sum_{v \in V} cpd(v) * total_distance(v)$$

Dans nos tests, nous considérons que les véhicules sont identiques en ce qui concerne le coût de déplacement $cpd(\cdot)$ et le facteur de tarification p . Nous avons fixé à $P = 1.5$, $p = 2$ et $cpd(v) = 1$ pour tout v .

1. <https://www.openstreetmap.org/>
2. <https://territoire.emse.fr/>

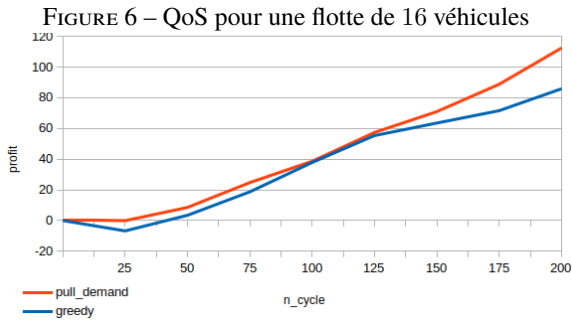
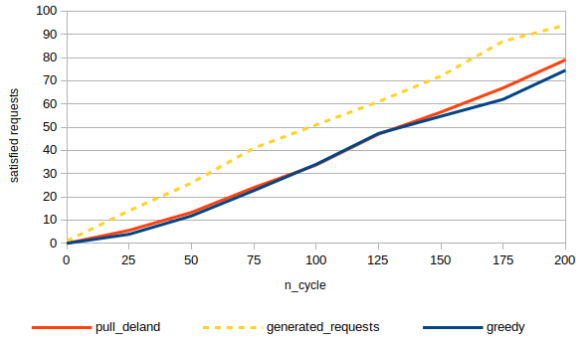


FIGURE 7 – QoB pour une flotte de 16 véhicules

6.2 Résultats expérimentaux

Pour évaluer la faisabilité de l’approche heuristique « *Pull-demand* », nous la comparons à une approche gloutonne (programmer une seule demande à l’avance) qui a été mentionnée par van Lon et al. [2012] comme la meilleure stratégie pour les réglages dynamiques, suite à une sélection génétique. Les deux approches comparées utilisent la même stratégie de sélection des demandes (fonction de priorité des demandes) au cours de chaque scénario. Les résultats présentés dans les Figures 6 et 7 concernent un scénario dans lequel les véhicules choisissent la demande la moins chère. La fonction de priorité retourne pour une demande d une valeur qui est inversement proportionnelle au coût de la demande pour v : $1/costDemand(d, v)$. Ainsi plus le coût est faible plus la priorité est élevée. $costDemand(d, v)$ est le coût marginal mentionné dans la section 4 ; il en va de même pour l’algorithme glouton, puisque la planification des véhicules ne peut contenir qu’une seule demande. Nous avons exécuté plusieurs instances de problèmes variant en fonction de la taille de la flotte n . Chaque instance de ces tests est exécutée 10 fois avec différentes graines aléatoires. Tout d’abord, nous évaluons avec une taille de flotte fixe $n = 16$ pour suivre la qualité de la solution dans le temps, puis avec une taille de flotte variable $n \in [4, 36]$ sur 200 cycles pour chaque scénario. Il est à noter qu’il est impossible d’atteindre un taux de qualité de service de 100% dans ces scénarios, car il y aura toujours des demandes qui seront générées jusqu’au dernier cycle, en supposant que l’exécution du scénario continue à les servir, alors que l’exécution s’arrête au dernier cycle, les laissant non servies. Bien que les valeurs de QoS de notre algorithme soient proches de

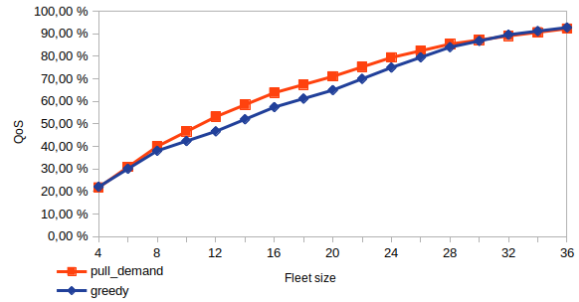


FIGURE 8 – QoS avec des flottes de tailles croissantes

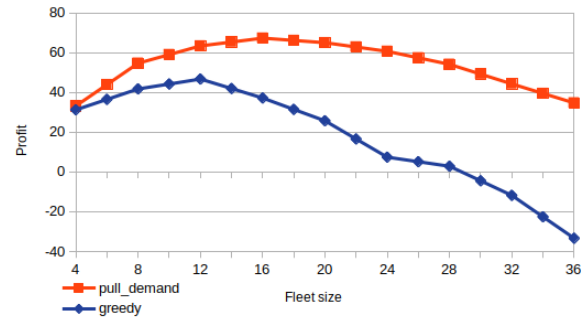


FIGURE 9 – QoB avec des flottes de tailles croissantes

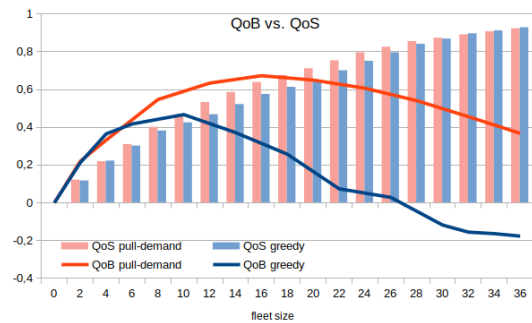


FIGURE 10 – Évolution de QoS vs. QoB

celles générées par l’approche gloutonne, elles restent légèrement supérieures à celles-ci dans la plupart des cas, comme le montre la Figure 8. En partant d’une petite flotte de 4 véhicules, l’heuristique « *Pull-demand* » permet d’obtenir des solutions avec les mêmes valeurs de QoS et de QoB que celles de l’algorithme glouton. En augmentant la taille de la flotte, plus de demandes peuvent être satisfaites, ce qui augmente la valeur de QoS et de QoB. Nous constatons que l’augmentation de ces valeurs pour l’heuristique est plus importante que celle de l’approche gloutonne. Cela peut s’expliquer par le fait que la planification des demandes améliore le profit du système dans un avenir prévisible, car elle réduit le processus de déplacement sans destination spécifique jusqu’à ce qu’une nouvelle demande soit choisie, ce qui est le comportement des véhicules dans l’algorithme glouton.

La valeur de la qualité de service continue de croître lorsque

la taille de la flotte augmente, jusqu'à atteindre un seuil n_{QoS} pour lequel l'ajout de nouveaux véhicules devient inutile, car toutes les demandes reçues dans le système maintenant ou à l'avenir (sauf celles qui sont reçues dans les derniers moments d'exécution) peuvent être servies avec la taille actuelle de la flotte. Il en va de même pour la QoB, mais la différence ici est que les véhicules ajoutés entraîneront des dépenses opérationnelles supplémentaires, ce qui entraîne une perte de valeur du bénéfice après avoir atteint son seuil de croissance n_{QoB} . Habituellement, n_{QoB} est inférieur à n_{QoS} , et nous pouvons observer un compromis entre l'amélioration de la QoS ou de la QoB.

D'après la Figure 9, nous constatons que la flotte gloutonne dispose de $n_{QoB}^{greedy} = 12$ alors que sa QoS n'est que d'environ 60%, tandis que $n_{QoB}^{pull-demand} = 16$ avec une QoS d'environ 70%. Notez que $n_{QoB}^{gloutonne} < n_{QoB}^{pull-demand}$ et même si elle diminue ensuite, la valeur de $n_{QoB}^{pull-demand}$ reste supérieure à $n_{QoB}^{gloutonne}$. Cela donne un plus large éventail d'options pour combiner les améliorations de la solution fournie (QoS et QoB). Les résultats ci-dessus montrent que dans tous les cas, la programmation de plusieurs demandes à l'avance avec le protocole d'optimisation « Pull-demand » donne de meilleurs résultats que la programmation d'une seule demande, tout en nécessitant un temps de calcul réduit, et aucune information globale.

7 Conclusion

Nous avons proposé dans cet article un protocole décentralisé pour l'échange de demandes, s'appuyant sur une heuristique d'insertion, pour allouer des demandes aux véhicules dans le contexte du transport à la demande dynamique. Nous montrons au travers des exemples que le protocole de demande d'échange de demandes peut être une amélioration prometteuse de la qualité des solutions. Afin d'évaluer la faisabilité du protocole proposé, nous avons évalué les résultats de notre technique sur des données synthétiques pour les taxis opérant dans la ville de Saint-Étienne, et avons montré qu'elle surpasse une approche gloutonne classique. Dans les travaux futurs, nous prévoyons d'évaluer l'efficacité, la performance, la robustesse et l'optimalité de cette heuristique par rapport à différentes approches, en simulant différents paramètres sur la distribution de l'information, les critères de décision et différents niveaux de dynamique du problème.

Références

N. A. Agatz, A. L. Erera, M. W. Savelsbergh, and X. Wang. Dynamic ride-sharing : A simulation study in metro atlanta. *Transportation Research Part B : Methodological*, 45(9) :1450 – 1464, 2011.

A. M. Campbell and M. Savelsbergh. Efficient insertion heuristics for vehicle routing and scheduling problems. *Transportation science*, 38(3) :369–378, 2004.

P. Cramton, Y. Shoham, and R. Steinberg. An overview

of combinatorial auctions. *ACM SIGecom Exchanges*, 7 (1) :3–14, Dec. 2007.

- P. Danassis, A. Filos-Ratsikas, and B. Faltings. Anytime Heuristic for Weighted Matching Through Altruism-Inspired Behavior. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 215–222, Macao, China, Aug. 2019.
- K. C. Dey, A. Rayamajhi, M. Chowdhury, P. Bhavsar, and J. Martin. Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network – Performance evaluation. *Transportation Research Part C : Emerging Technologies*, 68 : 168–184, July 2016.
- M. Egan and M. Jakob. Market Mechanism Design for Profitable On-Demand Transport Services. *arXiv :1501.01582 [cs]*, Jan. 2015.
- M. El Falou, M. Itmi, S. El Falou, and A. Cardon. On demand transport system's approach as a multi-agent planning problem. In *2014 International Conference on Advanced Logistics and Transport (ICALT)*, pages 53–58. IEEE, 2014.
- A. Glaschenko, A. Ivaschenko, G. Rzevski, and P. Skobelev. Multi-Agent Real Time Scheduling System for Taxi Companies. *AAMAS*, page 8, 2009.
- J. M. S. Grau and M. A. E. Romeu. Agent Based Modeling for Simulating Taxi Services. *Procedia Computer Science*, 52 :902–907, 2015.
- J. M. S. Grau, M. A. E. Romeu, E. Mitsakis, and I. Stamos. Agent based modeling for simulation of taxi services. *Journal of Traffic and Logistics Engineering*, 1(2) :159–163, 2013.
- G. Picard, F. Balbo, and O. Boissier. Approches multi-agents pour l'allocation de courses à une flotte de taxis autonomes. In *RIA2018*, number 2 in *Revue d'intelligence artificielle*, pages 223–247, 2018.
- W. Shen and C. Lopes. Managing Autonomous Mobility on Demand Systems for Better Passenger Experience. In *PRIMA 2015 : Principles and Practice of Multi-Agent Systems*, volume 9387, pages 20–35, Cham, 2015. Springer International Publishing.
- M. M. Solomon. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, 35(2) :254–265, 1987.
- R. R. van Lon, T. Holvoet, G. Vanden Berghe, T. Wenseleers, and J. Branke. Evolutionary synthesis of multi-agent systems for dynamic dial-a-ride problems. In *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion - GECCO Companion '12*, page 331, Philadelphia, Pennsylvania, USA, 2012. ACM Press.

SANDMAN : un système auto-adaptatif pour la détection d'anomalies dans le flux des données des bâtiments intelligents

Maxime Houssin^{1,2}, Stéphanie Combettes¹, Marie-Pierre Gleizes¹, Berangere Lartigue²
Université Toulouse 3 Paul Sabatier : IRIT¹, LMDC²
prenom.nom@{irit.fr¹, univ-tlse3.fr²}

Résumé

Actuellement, la gestion de l'énergie au sein des bâtiments est essentielle pour participer à la transition écologique. Pour cela, les bâtiments sont de plus en plus équipés de capteurs pour aider le gestionnaire du bâtiment. Mais l'hétérogénéité et la grande quantité de données générées rend sa tâche assez ardue. Le système multi-agent SANDMAN, décrit dans cet article, a pour objectif d'aider à la détection automatique en temps réel, de plusieurs types d'anomalies en utilisant des données brutes et hétérogènes. SANDMAN fait un apprentissage semi-supervisé en considérant quelques avis d'un expert du domaine. Les résultats montrent que SANDMAN, après une phase d'apprentissage, détecte les différents types d'anomalies, est résistant au bruit et passe l'échelle.

Mots Clef

Système multi-agent auto-adaptatif, Détection d'anomalies, Bâtiments intelligents

Abstract

Currently, energy management within buildings is essential to participate in the green transition. To this end, buildings are increasingly equipped with sensors to assist the building manager. But the heterogeneity and the large amount of data generated makes this task quite difficult. The SANDMAN multi-agent system, described in this paper, aims to assist in the automatic detection, in real time, of several types of anomalies using raw and heterogeneous data. SANDMAN features a semi-supervised learning by considering some feedbacks from an expert in the field. The results show that SANDMAN, after a learning phase, detects the different types of anomalies, is resistant to noise and is scalable.

Keywords

Self-Adaptive Multi-Agent System, Anomaly Detection, Smart Buildings

1 Introduction

Les bâtiments représentent plus de 20% de la consommation mondiale d'énergie, ce chiffre pouvant dépasser 40 % dans les pays développés. Cependant, une grande partie de cette énergie est gaspillée [1] (30 % aux États-Unis et

en Europe). Ce gaspillage provient d'une mauvaise gestion des bâtiments comme la non-détection de problèmes.

L'objectif du travail présenté dans cet article concerne la détection d'anomalies dans les données énergétiques au sein de bâtiments intelligents pour mettre en place rapidement des mécanismes pour lutter contre ce gaspillage d'énergie.

Avec l'avènement de l'IoT (*Internet of Things*), le nombre de capteurs dans les bâtiments existants et nouveaux a fortement augmenté en raison de leur coût moins élevé et de l'avantage évident de leur utilisation pour la gestion des bâtiments. Des capteurs peuvent être facilement ajoutés dans les bâtiments ou remplacés par d'autres. Par conséquent, la gestion de ces capteurs et des données qu'ils génèrent font que les gestionnaires des bâtiments se trouvent face à un système complexe à gérer.

Un contrôle précis de ces données est pourtant nécessaire pour gérer correctement les bâtiments, et notamment leur performance énergétique. C'est pourquoi un outil d'aide à la détection automatique des anomalies est un atout important pour les gestionnaires de bâtiments. Cet outil doit traiter les données **en temps réel** pour pouvoir agir au plus tôt, ceci étant un enjeu essentiel dans la détection des anomalies. Parce que le nombre de données est important, les espaces de recherche des anomalies deviennent aussi très grands, c'est pourquoi le système doit apprendre à lever les anomalies de manière semi-supervisée grâce aux feedbacks d'un expert.

Afin d'établir un état de l'art adapté à notre problématique, nous listons au préalable les caractéristiques qui nous semblent indispensables pour un système de détection d'anomalies, ainsi que les définitions des différents types d'anomalies.

1.1 Caractéristiques requises pour un système de détection d'anomalies

Un bâtiment intelligent équipé de capteurs engendre une grande quantité de données disponibles qui doivent être analysées pour améliorer la gestion d'énergie. Par exemple, le SGE (Service de Gestion et d'Exploitation de l'énergie pour le campus de l'Université P. Sabatier - Toulouse III) gère environ 6000 capteurs mesurant au moins une valeur par heure. Ainsi, la conception de systèmes intelligents de détection d'anomalies dans les bâtiments doit prendre en

compte les caractéristiques suivantes :

- **Interaction avec un expert.** Afin d'informer l'expert sur les anomalies trouvées par le système mais aussi pour avoir des feedbacks de l'expert pour permettre au système d'apprendre tout le long de sa vie.
- **Détection en temps réel.** Afin de réduire l'impact énergétique des anomalies et éventuellement l'inconfort pour les occupants, il est préférable de détecter au plus tôt les anomalies.
- **Détection de plusieurs types d'anomalies.** Pour être le plus exhaustif possible, le système de détection d'anomalies doit être capable de détecter différents types d'anomalies comme les anomalies dues à un seul capteur, des anomalies dues à plusieurs capteurs, des anomalies dues à l'évolution de valeurs de capteurs sur une période donnée et donc qui ne se manifestent qu'au bout d'un certain temps, etc.
- **Utilisation de données brutes.** Compte tenu de la grande quantité de données à manipuler et de la nécessité d'un traitement en temps réel, il convient d'éviter autant que possible le prétraitement des données. En effet, c'est une tâche chronophage, qui peut amener des biais. De plus, le traitement en temps réel laisse peu de temps pour formater les données au fur et à mesure de leur arrivée.
- **Utilisation de données hétérogènes.** Les capteurs au sein d'un bâtiment intelligent mesurent différentes grandeurs physiques (température, pression, consommation d'énergie, etc.) qui doivent toutes être prises en compte pour une meilleure détection des anomalies.
- **Ouverture.** Dans un bâtiment intelligent, des capteurs peuvent être ajoutés et d'autres enlevés. Le système de détection d'anomalies doit continuer à fonctionner dans ces environnements dynamiques.
- **Passage à l'échelle.** Le système de détection d'anomalies doit pouvoir s'appliquer non seulement aux bâtiments, mais également à des ensembles de bâtiments, et à terme, à des campus, des quartiers. Le nombre de capteurs à gérer peut donc être considérable.
- **Généricité.** Pour être utilisable de manière optimale, le système de détection d'anomalies doit pouvoir être déployé dans plusieurs bâtiments sans modification majeure. De plus, dans le domaine des bâtiments intelligents, il existe des systèmes de détection d'anomalies pour le chauffage, d'autres pour la consommation d'électricité, etc. Le développement d'un système unique pour détecter les anomalies dans ces différents domaines est un avantage.

Le système SANDMAN utilise une technique d'apprentissage couplée à l'utilisation de systèmes multi-agents pour prendre en compte ces caractéristiques. Par ailleurs, ce sys-

| row | timestamp | sensor1 | | sensor2 | | sensor3 | | sensor4 | | sensor5 | | anomaly |
|-----|------------------|----------|---------|----------|---------|----------|---------|----------|---------|----------|---------|---------|
| | | measured | nominal | measured | nominal | measured | nominal | measured | nominal | measured | nominal | |
| 0 | 2016-01-11T00:00 | 16 | 17 | 11 | 10 | 160 | 160 | 11 | 11 | 66 | 66 | |
| 1 | 2016-01-11T01:00 | 15 | 15 | 11 | 11 | 140 | 140 | -5 | 11 | 64 | 65A | |
| 2 | 2016-01-11T02:00 | 15 | 15 | 12 | 12 | 130 | 127 | 10 | 10 | 65 | 65 | |
| 3 | 2016-01-11T03:00 | 15 | 16 | 12 | 13 | 110 | 110 | 8 | 8 | 64 | 65 | |
| 4 | 2016-01-11T04:00 | 15 | 14 | 12 | 12 | 120 | 120 | 9 | 9 | 65 | 65 | |
| 5 | 2016-01-11T05:00 | 15 | 15 | 14 | 14 | 140 | 140 | 13 | 13 | 400 | 65 | |
| 6 | 2016-01-11T06:00 | 16 | 17 | 16 | 14 | 170 | 165 | 17 | 17 | 66 | 66 | |
| 7 | 2016-01-11T07:00 | 17 | 19 | 20 | 20 | 220 | 200 | 25 | 25 | 67 | 67 | |
| 8 | 2016-01-11T08:00 | 23 | 19 | 19 | 23 | 220 | 235 | 34 | 29 | 68 | 67A | |
| 9 | 2016-01-11T09:00 | 18 | 18 | 25 | 25 | 260 | 255 | 33 | 33 | 68 | 68 | |
| 10 | 2016-01-11T10:00 | 19 | 18 | 27 | 27 | 230 | 245 | 35 | 34 | 69 | 69 | |
| 11 | 2016-01-11T11:00 | 19 | 19 | 25 | 25 | 210 | 235 | 32 | 32 | 69 | 70 | |
| 12 | 2016-01-11T12:00 | 21 | 21 | 26 | 24 | 210 | 230 | 32 | 32 | 71 | 71 | |
| 13 | 2016-01-11T13:00 | 22 | 22 | 28 | 28 | 220 | 240 | 34 | 34 | 72 | 72A | |
| 14 | 2016-01-11T14:00 | 22 | 22 | 29 | 29 | 270 | 270 | 34 | 33 | 72 | 72 | |
| 15 | 2016-01-11T15:00 | 23 | 23 | 31 | 31 | 270 | 270 | 35 | 35 | 73 | 73 | |
| 16 | 2016-01-11T16:00 | 24 | 24 | 32 | 29 | 270 | 270 | 35 | 35 | 76 | 74 | |
| 17 | 2016-01-11T17:00 | 23 | 23 | 29 | 29 | 250 | 258 | 31 | 31 | 73 | 73 | |
| 18 | 2016-01-11T18:00 | 22 | 24 | 29 | 29 | 240 | 240 | 31 | 30 | 72 | 72 | |
| 19 | 2016-01-11T19:00 | 21 | 22 | 26 | 26 | 220 | 220 | 27 | 27 | 71 | 71 | |
| 20 | 2016-01-11T20:00 | 19 | 19 | 22 | 22 | 200 | 200 | 23 | 23 | 69 | 69 | |
| 21 | 2016-01-11T21:00 | 19 | 20 | 21 | 22 | 190 | 185 | 21 | 19 | 70 | 69 | |
| 22 | 2016-01-11T22:00 | 18 | 18 | 20 | 20 | 180 | 180 | 20 | 20 | 68 | 68 | |
| 23 | 2016-01-11T23:00 | 17 | 17 | 16 | 16 | 170 | 170 | 16 | 16 | 67 | 67 | |

TABLE 1 – Exemple d'une base de données

tème lève des anomalies à destination d'un expert qui doit réagir au mieux. Le système et l'expert sont en interaction : l'expert doit pouvoir voir les levées d'anomalies mais il doit également pouvoir confirmer/infirmier ces anomalies ou en signaler des nouvelles si nécessaire à SANDMAN. Le système SANDMAN est donc un système d'apprentissage semi-supervisé puisqu'il peut recevoir des feedbacks d'un expert. Précisons toutefois que l'expert peut (in)valider une levée d'anomalie *a posteriori* ; il n'est pas tenu d'être constamment actif devant le système et l'apprentissage du système prend en compte cette spécificité.

1.2 Définition des différents types d'anomalies

Chaque capteur fournit une donnée estampillée avec l'heure de sa perception ou *timestamp*.

Pour chacun, on dispose d'une **valeur réelle** qui correspond à la mesure effectuée et une **valeur nominale** qui est la valeur habituellement fournie par le capteur. Cette valeur nominale est calculée à partir d'un profil. La localisation des capteurs est inconnue, ainsi que toute autre méta-donnée (type du capteur, fréquence d'acquisition, etc.). Comme le manque d'information sur les capteurs est un problème courant [2], il est nécessaire de les prendre tous en compte dans la méthode de détection des anomalies.

Comme Chandola et al. [3], nous considérons qu'une anomalie est une situation inattendue ou indésirable dans un système. Ainsi trois types d'anomalies peuvent être pris en compte. Une anomalie est **ponctuelle** si une valeur réelle est en dehors de la plage de valeur acceptable pour le capteur ; une anomalie est **contextuelle** si une valeur réelle se situe dans une plage acceptable pour le capteur mais est anormale dans certains contextes (exemple : consommation de chauffage pendant l'été) ; une anomalie est **collective** si un ensemble de valeurs réelles est anormal par rapport à l'ensemble des données, bien que les valeurs réelles individuelles ne soient pas anormales.

Le tableau 1 illustre les 3 types d'anomalies définies ci-dessus. Elles sont indiquées par la lettre A dans la dernière colonne. La ligne 2 est considérée comme anormale par l'expert parce que le capteur 4 a une valeur de -5, ce qui est éloi-

gné de la valeur nominale de 11. Il s'agit d'une anomalie ponctuelle.

La ligne 8 est une anomalie à cause des valeurs des capteurs 1 à 4. Bien que prise individuellement, aucune de ces valeurs ne serait considérée comme une anomalie, toutes ensemble elles génèrent une anomalie collective.

La dernière anomalie de la ligne 13 apparaît à 13 heures et seulement à ce moment-là. Elle est due au fait que les valeurs réelles du capteur 3 sont disparates de la ligne 9 à la ligne 13. Dans la vie réelle, un expert peut avoir remarqué un problème à 13 heures en raison de l'accumulation constante de petites différences dans les 4 lignes précédentes. C'est ce qu'on appelle une anomalie contextuelle car il est nécessaire de prendre en compte le passé récent pour la détecter.

La ligne 5 n'est pas une anomalie même si la valeur de 400 du capteur 5 est très éloignée de la valeur nominale qui est de 65. Elle n'est pas considérée comme une anomalie par l'expert, ce qui signifie que ce capteur est soit sans importance pour la détection des anomalies, soit que la variation n'est pas assez grande pour qu'un expert la considère comme anormale.

L'article se compose de la façon suivante. La section 2 décrit l'état de l'art des principales méthodes existantes de détection des anomalies. La section 3 explicite l'architecture du système multi-agent proposé, SANDMAN (*semi-Supervised ANomaly Detection with Multi-AgeNt systems*). La section 4 est consacrée aux expérimentations et aux résultats de SANDMAN. Enfin, la section 5 conclut et propose des perspectives au travail présent.

2 Etat de l'art sur la détection d'anomalies dans les bâtiments intelligents

Compte tenu de l'enjeu réel de la détection des anomalies dans les bâtiments, le champ de recherche dans ce domaine est en pleine expansion. Les systèmes multi-agents ont été introduits en physique du bâtiment depuis une dizaine d'années, mais leurs applications est quasiment entièrement dédiées à l'optimisation de la gestion des systèmes (chauffage, refroidissement, ventilation) ou du bâtiment complet [4]. Aucune étude, à notre connaissance, n'a été menée pour une application dans la détection des anomalies dans les données des bâtiments.

Parmi les méthodes utilisées dans la littérature, on trouve les **modèles physiques**, qui nécessitent de modéliser le système ou bâtiment et de le comparer avec les données mesurées. Ces méthodes sont vite limitées, compte tenu de la complexité des systèmes à modéliser [5].

Des méthodes de **classification non supervisée et statistiques** ont été utilisées pour la détection des anomalies dans des données de bâtiment [6] [7] [8]. Cependant, ces études ne portent que sur un faible nombre de données et ne montrent pas que le passage à l'échelle soit possible. Elles reposent également sur l'hypothèse que le nombre d'ano-

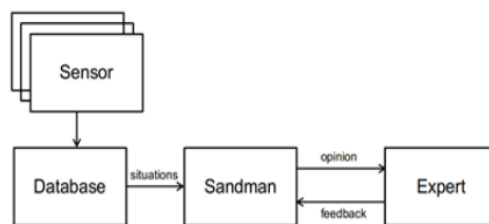


FIGURE 1 – SANDMAN : schéma général

malies est faible par rapport au nombre total de données.

Les **réseaux de neurones** sont des outils largement utilisés dans les études sur les bâtiments, mais rarement pour la détection des anomalies [9], [10]. Même si les réseaux de neurones peuvent détecter des anomalies en temps réel, un grand ensemble de données étiquetées est nécessaire pour entraîner les réseaux. Or, cette étape semble impossible à mener pour les tailles de systèmes considérés ici et elle ne correspond pas à ce qui se passe réellement dans la gestion des bâtiments.

Les méthodes de **data mining** sont utilisées pour la détection d'anomalies dans des données de bâtiments [11], [12], [13]. Cependant, une étape de pré-traitement des données est nécessaire, ainsi qu'un expert pour la sélection des données pertinentes lors du passage à l'échelle.

Enfin, des méthodes de **systèmes décentralisés avec multi-agents** ont été utilisées pour la détection d'anomalies dans des données estampillées avec l'heure de leur perception [14][15]. Ces méthodes n'étaient pas appliquées aux bâtiments intelligents, mais pourraient y être adaptées.

3 Architecture de SANDMAN

Cette section présente les principes de base de SANDMAN. La première sous-section décrit les notions de situation et de profil des capteurs, la deuxième explique le fonctionnement du système SANDMAN pour détecter des anomalies et enfin la dernière explicite l'apprentissage. La figure 1 fait référence au fonctionnement général de SANDMAN.

3.1 Définitions : situation et profil d'un capteur

SANDMAN travaille à partir d'informations remontées par un ensemble de capteurs donnés. Par exemple, dans le tableau 1, SANDMAN a accès aux valeurs de 5 capteurs. Nous avons défini la notion de **situation** comme l'ensemble des valeurs mesurées de tous ces capteurs sur 24 heures et on nomme **situation courante** la dernière situation rencontrée. La dernière mesure de tous les capteurs est appelée **valeurs instantanées des capteurs** : ceci correspond dans le tableau 1 à la dernière ligne, contenant les valeurs instantanées des 5 capteurs à 23H00. Les situations rencontrées par SANDMAN sont mémorisées dans un historique des situations.

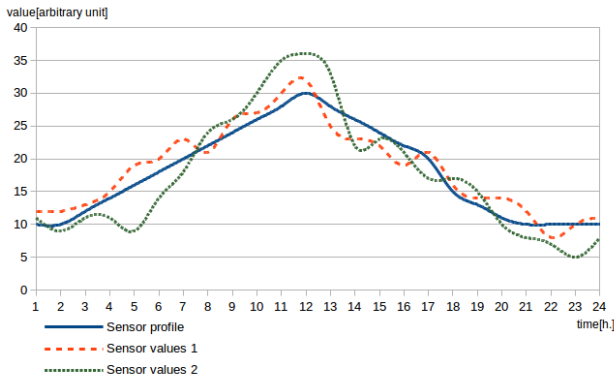


FIGURE 2 – Valeurs mesurées du capteur et profil du capteur en fonction du temps

A chaque capteur est associé un **profil**, constitué de 24 valeurs mesurées, une pour chaque heure de la journée. Ce profil indique la valeur attendue à tout moment de la journée pour le capteur considéré. Il est mis à jour par SANDMAN tout au long de la vie du système.

La figure 2 représente le profil de ce capteur en fonction du temps sur une période de 24 heures et on a représenté par les valeurs mesurées *values 1* un exemple possible de valeurs mesurées pour ce capteur et les valeurs mesurées *values 2* un autre exemple possible de valeurs mesurées pour ce capteur. Les valeurs mesurées *values 1* et *values 2* sont ici deux exemples distincts, une seule valeur pouvant être mesurée à la fois sur un capteur. On peut constater que les valeurs mesurées *values 1* sont proches du profil attendu sur 24 heures alors que les valeurs mesurées *values 2* sont considérées comme anormales.

3.2 Détection d'anomalies par SANDMAN

Etant donné que tous les capteurs disponibles sont utilisés, que l'expert peut ne s'intéresser qu'à certaines anomalies, et que chaque capteur a un intervalle et une amplitude de valeurs qui lui est propre, l'apprentissage des profils ne suffit pas pour détecter les anomalies. En effet, certains capteurs peuvent ne pas être pertinents pour la détection des anomalies. C'est le cas, par exemple, de capteurs situés dans un bâtiment qui n'intéresse pas l'expert, ou bien si l'expert ne s'intéresse qu'à la consommation électrique alors un capteur de qualité de l'air ne sera pas pertinent. Enfin, pour les capteurs qui sont effectivement utiles, selon les capteurs, certains écarts par rapport aux valeurs attendues sont plus importants que d'autres. Ceci est dû au fait que leurs valeurs ne sont pas normalisées car SANDMAN utilise des données brutes.

Aussi, pour lever ou non une anomalie à chaque nouvelle situation courante, SANDMAN utilise à la fois les valeurs mesurées des capteurs sur 24h et les profils des capteurs associés. Pour cela, SANDMAN calcule **la disparité d'un capteur** qui est la somme de toutes les différences entre le profil du capteur et les valeurs mesurées du capteur au

cours des dernières 24 heures, c'est-à-dire pour la situation courante (Eq. (1)).

$$Disparity_s^t = \sum_{t_i=t}^{t-23} |realValue_s^{t_i} - nominalValue_s^{t_i}| \quad (1)$$

avec :

- c : capteur c ;
- t : instant t de la situation courante ;
- $realValue_c^l$: valeur réelle du capteur c au temps l ;
- $nominalValue_c^l$: valeur nominale du capteur c au temps l ;

La période considérée est une fenêtre glissante de 24 heures. Si l'heure de la situation courante est 15h, toutes les données de 16h la veille à 15h le jour courant sont considérées pour le profil allant de 16h à 15h. SANDMAN calcule ensuite **le degré d'anomalie (DA) de la situation courante** qui correspond à la somme des valeurs mesurées instantanées de tous les capteurs, pondérées par un poids (Eq. (2)).

$$DA(Situation^t) = \sum_{c=1}^S Disparity_c^t * Poids_c \quad (2)$$

avec :

- $Situation^t$: situation au temps t ;
- S : nombre de capteurs ;
- $Disparity_c^t$: disparité du capteur c calculée au temps t ;
- $Poids_c$: poids associé au capteur c .

Ainsi grâce au degré d'anomalie, SANDMAN peut classer la situation courante, c'est-à-dire de l'étiqueter "normale" ou "anormale" en suivant l'algorithme 1.

Algorithme 1 Algorithme de classification ou de détection d'anomalies

- 1: **pour** chaque nouvelle situation courante **faire**
 - 2: **si** le degré d'anomalie de la situation courante est inférieur à un seuil **alors**
 - 3: SANDMAN retourne : "la situation est normale"
 - 4: **sinon**
 - 5: SANDMAN retourne : "la situation est anormale"
 - 6: **fin si**
 - 7: **fin pour**
-

Pour détecter les anomalies de manière correcte, il faut donc que les poids $Poids_i$ associés aux capteurs aient les bonnes valeurs. Un capteur a un seul poids associé qui est utilisé pour calculer le degré d'anomalie pour toutes les situations de l'historique des situations.

3.3 Apprentissage par SANDMAN

Une fois que SANDMAN a effectué la classification d'une situation c'est-à-dire qu'il l'a étiquetée "normale" ou "anormale" et qu'il a un retour de l'expert, il effectue un cycle de résolution au cours duquel il analyse s'il doit ou non apprendre. SANDMAN analyse donc les cas suivants :

1. Pour une situation étiquetée "anormale" pour lesquelles SANDMAN et l'expert sont d'accord,
 - (a) SANDMAN ne fait rien.
2. Pour une situation étiquetée "anormale" pour l'expert et "normale" pour SANDMAN,
 - (a) SANDMAN ajoute à l'historique des situations cette situation étiquetée "anormale"
 - (b) SANDMAN lance l'auto-adaptation des poids
3. Pour chaque ligne de la situation étiquetée "normale" pour l'expert et "anormale" pour SANDMAN,
 - (a) SANDMAN met à jour les profils des capteurs
 - (b) SANDMAN ajoute à l'historique des situations cette situation étiquetée "normale"
 - (c) SANDMAN lance l'auto-adaptation des poids
4. Pour chaque ligne de la situation étiquetée "normale" pour l'expert et pour SANDMAN,
 - (a) SANDMAN met à jour les profils des capteurs

Les principales étapes concernant la mise à jour des profils, l'ajout d'une situation dans l'historique et l'auto-adaptation des poids sont décrits ci-après.

Mise à jour des profils. Ils ne sont mis à jour que si la situation est étiquetée "normale" par l'expert (étapes 3a et 4a), car les valeurs des capteurs dans les situations anormales ne sont pas fiables et ne représentent pas les valeurs attendues. La formule de mise à jour appliquée à chaque profil de capteur est la suivante (Eq. 3) :

$$val_{profil}^{tt} = (1 - \lambda) * val_{profil}^t + \lambda * val_{capteur}^t \quad (3)$$

avec

- t : l'heure de la situation courante,
- $\lambda \in]0, 1[$ représente l'importance de la nouvelle valeur par rapport à l'ancienne,
- val_{profil}^t : la valeur précédemment apprise et stockée dans le profil pour l'heure t
- $val_{capteur}^t$: la valeur mesurée du capteur pour l'heure t
- val_{profil}^{tt} : la nouvelle valeur apprise et stockée dans le profil pour l'heure t

Chaque profil de capteur modifie l'une de ses 24 valeurs correspondant à l'heure t de la situation courante en utilisant la valeur mesurée $val_{capteur}^t$ de la situation courante.

Ajout d'une situation à l'historique des situations. Cet ajout, réalisé aux étapes 2a et 3b, lance des cycles de résolution de SANDMAN qui se produisent jusqu'à ce que

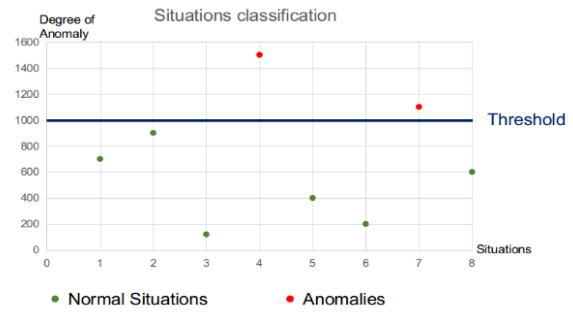


FIGURE 3 – Example of 8 classified situations

toutes les situations présentes dans l'historique soient classées correctement et de manière équilibrée. Une situation est classée correctement si son degré d'anomalie permet de déduire le bon classement (normal ou anormale) de la situation. Les situations sont classées de manière équilibrée si le degré d'anomalie de la pire situation étiquetée "normale" et le degré d'anomalie de la pire situation étiquetée "anormale" sont en valeur absolue à une même distance du seuil donné. Ceci est illustré sur la figure 3, où les points respectivement au dessus et au-dessous du seuil *Threshold* et les plus près du seuil sont à égale distance de ce seuil.

A chaque situation est attribuée une valeur de criticité en fonction de son degré d'anomalie et de sa classification donnée par l'expert. La criticité représente le degré de satisfaction de la situation par rapport à son objectif. Une situation représentant une situation étiquetée "normale" vise un degré d'anomalie aussi bas que possible par rapport au seuil *Threshold*, tandis qu'une situation étiquetée "anormale" vise un degré d'anomalie élevé par rapport au seuil *Threshold*. La criticité **Crit** est calculée comme suit :

$$Crit_{Situation^t} = DA(Situation^t) - Threshold \quad (4)$$

pour une situation "normale".

$$Crit_{Situation^t} = Threshold - DA(Situation^t) \quad (5)$$

pour une situation "anomalie".

Chaque situation veut minimiser sa criticité, une criticité négative signifie que la situation est correctement classée. Comme les disparités et les poids sont toujours positifs, les situations étiquetées "normales" et les situations étiquetées "anormales" ont des objectifs antagonistes, voulant respectivement des pondérations plus ou moins élevées. Lorsqu'un cycle de résolution est effectué, chaque situation envoie un message à chaque agent poids avec les informations suivantes :

- le sens de changement de poids (plus bas pour une situation étiquetée "normale", plus haut pour une situation étiquetée "anormale");

- la criticité de la situation
- l'influence du poids dans le degré d'anomalie.

L'influence d'un poids associé aux capteurs dans une situation correspond à la part du capteur dans le calcul du degré d'anomalie. Elle est calculée comme suit :

$$Influ_w = \frac{Poids_c * Disparity_c^t}{DA(Situation^t)} \quad (6)$$

Un cycle de résolution débute quand les situations demandent aux poids de s'ajuster, le cycle se termine quand les poids se sont auto-adaptés.

Auto-adaptation des poids. Chaque poids est représenté par un agent poids. Le système multi-agent des poids a comme objectif que chaque agent poids trouve sa valeur en coopérant avec les autres. Au cours d'un cycle de résolution, toutes les situations demandent des ajustements de poids (étapes 2b et 3c) et les agents poids s'auto-adaptent comme décrit dans l'algorithme 2, puis chaque situation calcule son nouveau degré d'anomalie en utilisant les poids mis à jour.

À la réception des paramètres envoyés par la situation : le sens, la criticité et l'influence, les agents poids décident de mettre à jour leur poids ou non, indépendamment et simultanément. Pour ce faire, ils suivent l'algorithme 2.

Algorithme 2 Auto-adaptation des agents poids

Entrée: *Sens, Crit, Influence* de toutes situations

- 1: Sélectionner la situation normale la plus critique *NSA* et la situation anormale la plus critique *ASA*
 - 2: **si** *Crit(NSA) > Crit(ASA)* **alors**
 - 3: **si** *Influ(NSA) > Influ(ASA)* **alors**
 - 4: l'agent poids diminue sa valeur
 - 5: **fin si**
 - 6: **si** *Influ(ASA) > Influ(NSA)* **alors**
 - 7: l'agent poids augmente sa valeur
 - 8: **fin si**
 - 9: **fin si**
-

Dans le tableau 1, la ligne 5 n'est pas une anomalie même si la valeur de 400 du capteur 5 est très éloignée de la valeur nominale qui est de 65. Elle n'est pas considérée comme une anomalie par l'expert, ce qui signifie que ce capteur est soit sans importance pour la détection des anomalies, soit que la variation n'est pas assez grande pour qu'un expert la considère comme anormale. La ligne 5 n'est pas une anomalie même si la valeur de 400 du capteur 5 est très éloignée de la valeur nominale qui est de 65. Elle n'est pas considérée comme une anomalie par l'expert, ce qui signifie que ce capteur est soit sans importance pour la détection des anomalies, soit que la variation n'est pas assez grande pour qu'un expert la considère comme anormale.

La coopération entre les agents poids garantit qu'au moins un agent poids actualise son poids et que chaque cycle d'actualisation des agents poids se traduit par un meilleur état général où l'agent poids le plus critique diminue sa criticité [16]. Après chaque cycle de mise à jour des

agents poids, les situations calculent leur nouvelle criticité en tenant compte des nouvelles valeurs des agents poids. Ce processus est répété jusqu'à ce que les situations les plus critiques étiquetées "normales" et "anormales" aient leurs criticités respectives égales et négatives. Cela garantit que toutes les situations sont correctement classées et que les situations suivantes ont de meilleures chances d'être classées en conséquence par SANDMAN. En effet, la meilleure façon de réduire le taux de faux négatifs et de faux positifs [3], et donc d'avoir un système robuste, est de maintenir les situations connues les plus critiques (sauvegardées dans l'historique) aussi loin du seuil que possible.

4 Expérimentations

Dans cette section, des expérimentations sont menées pour montrer la capacité de SANDMAN à détecter plusieurs types d'anomalies en utilisant les feedbacks d'un expert.

4.1 Description du cadre expérimental

Les données utilisées pour mener à bien les expérimentations ont été générées à l'aide du générateur de séries temporelles TSimulus [17]. Ces séries temporelles comportent une valeur mesurée signée pour chaque capteur pour chaque heure. La plage de valeurs et l'amplitude de variation est spécifique à chaque capteur. Les valeurs pour chaque capteur sont cycliques, avec ou sans bruit sur une période de 24 heures, c'est-à-dire que la valeur d'un capteur à 15 heures est la même chaque jour (au bruit près). Ces données synthétiques sont modifiées par un expert humain afin d'y introduire des anomalies de différents types. SANDMAN traite les données en temps réel et l'expert donne son avis a posteriori de manière asynchrone.

Toutes les expérimentations ont été réalisées sur un processeur à 4 coeurs d'une fréquence de 2,6 GHz. Les données utilisées dans chaque expérimentation ont des mesures toutes les heures pendant un mois, pour un total de 744 heures. 4 fichiers de données distincts sont utilisés pour montrer la capacité de SANDMAN à : i) détecter les anomalies ponctuelles et pallier le bruit, ii) détecter des anomalies collectives et contextuelles, iii) adapter les profils des capteurs, iv) passer à l'échelle.

Dans les expériences, nous avons choisi de reproduire l'ensemble des données d'un mois afin de montrer l'amélioration apportée par l'apprentissage du SANDMAN. Ainsi, les résultats sont présentés après un mois et deux mois.

Les résultats présentés dans cette section sont le résultat de ces deux mois. Les résultats sont présentés sous la forme du nombre de :

- VN : Vrai Négatif, VP : Vrai Positif,
- FN : Faux Négatif, FP : Faux Positif,
- *t/sit* : temps de calcul par situation traitée en ms.

4.2 Anomalies ponctuelles et résilience au bruit

Les anomalies ponctuelles sont le type d'anomalie le plus souvent détecté dans la littérature, car les approches univa-

riées et multivariées peuvent les détecter.

Dans notre expérimentation, les données mesurées sont bruitées comme le sont les données issus de capteurs vieillissant ou ayant une précision limitée. La base de données contenant des anomalies ponctuelles est utilisée pour étudier l'effet du bruit des données sur la précision de la détection des anomalies et les performances du système. Le fichier de données contient 20 capteurs, soit $20 \times 744 = 14880$ données. 58 anomalies ponctuelles ont été ajoutées manuellement par un expert. Le bruit est simulé en appliquant une distribution uniforme à chaque capteur avec la formule suivante :

$$Valeur_{bruit} = Valeur_{sansBruit} - \frac{intervalle}{2} + rand_{[0,1]} * intervalle * bruit$$

Avec :

- *intervalle* : $|MIN(c) - MAX(c)|$ pour chaque capteur c
- *bruit* : la quantité de bruit, respectivement 1% et 5% dans les expérimentations.

Le tableau 2 montre qu'un niveau faible de bruit n'a pas d'impact sur le fonctionnement de SANDMAN. Cependant un niveau de bruit plus élevé mène à un plus fort taux de faux positifs. SANDMAN peut réduire ce taux de faux positifs en créant des situation normales et ainsi apprendre des poids plus bas.

4.3 Anomalies collectives et contextuelles

Dans cette expérimentation, la base de données contient des anomalies ponctuelles, contextuelles et collectives. Les anomalies contextuelles surviennent lorsqu'une valeur de capteur est inhabituelle plusieurs fois de suite. Les anomalies collectives surviennent lorsque plusieurs valeurs de capteur pour la même heure sont inhabituelles, mais pas assez pour qu'une seule d'entre elles provoque une anomalie ponctuelle.

Le tableau 3 donne le nombre de chaque type d'anomalie, ainsi que le nombre de vrais positifs et faux négatifs au premier mois, puis au deuxième mois. Il n'y a pas de faux positifs dans les résultats de cette expérimentation. SANDMAN est capable de créer et d'ajouter à l'historique les situations correspondant aux situations mal classifiées au mois 1 pour les classifier sans erreur au mois 2, et ce, quelque soit le type d'anomalie.

4.4 Mise à jour du profil sur des données glissantes

Dans les expérimentations précédentes, le profil de chaque capteur a été stocké et a contribué à la détection des anomalies, mais les données étaient cycliques sur une base journalière. Dans cette expérimentation, les valeurs mesurées des capteurs sont modifiées de manière à ce que chaque valeur mesurée soit supérieure de 1% chaque jour, de sorte que sur un mois de 31 jours, chaque capteur a des valeurs mesurées supérieures de 31% aux données de contrôle. Le fichier de

| Bruit | Mois 1 | | | | Mois 2 | | | |
|--------------|--------|-----|----|----|--------|-----|----|----|
| | TP | TN | FN | FP | TP | TN | FN | FP |
| Pas de Bruit | 40 | 686 | 18 | 0 | 58 | 686 | 0 | 0 |
| 1% | 40 | 686 | 18 | 0 | 58 | 686 | 0 | 0 |
| 5% | 58 | 655 | 0 | 31 | 58 | 679 | 0 | 7 |

TABLE 2 – Résultats de la détection d'anomalies ponctuelles avec bruit

| | nb | Mois 1 | | Mois 2 | |
|--------------|----|--------|----|--------|----|
| | | TP | FN | TP | FN |
| Ponctuelle | 6 | 2 | 4 | 6 | 0 |
| Collective | 5 | 1 | 4 | 5 | 0 |
| Contextuelle | 3 | 1 | 2 | 3 | 0 |
| Total | 14 | 4 | 10 | 14 | 0 |

TABLE 3 – Résultats de la détection des 3 types d'anomalies

test comporte 58 anomalies ponctuelles et chacun des 20 capteurs est la cause d'une anomalie au moins une fois. Le tableau 4 présente les résultats de l'expérimentation. La mise à jour du profil des capteurs est suffisamment réactive pour permettre à SANDMAN de classifier les anomalies quand les valeurs mesurées des capteurs changent au cours du temps.

4.5 Passage à l'échelle

Dans cette expérimentation, un nombre croissant de capteurs est utilisé pour mesurer l'effet d'un nombre de capteurs plus important sur le temps de calcul. Pour ce faire, un ensemble de données sur 20 capteurs distincts est dupliqué pour obtenir jusqu'à 800 capteurs. Les données utilisées sont les mêmes que pour l'expérience précédente sur les données glissantes sans la modification des valeurs de 1% par jour. Le tableau 5 présente le temps de calcul par situation en fonction du nombre de capteurs. Le temps de calcul au mois 2 est toujours plus court que celui du mois 1 car l'apprentissage des poids est effectué uniquement en mois 1. On remarque également que le temps de résolution est proportionnel au nombre de capteurs utilisés et que la différence de temps entre les deux mois est constante. Ceci est dû au fait que l'essentiel du temps d'exécution vient de la lecture des données brutes à partir d'une base de don-

| | Mois 1 | | | | Mois 2 | | | |
|--------------------|--------|-----|----|----|--------|-----|----|----|
| | TP | TN | FN | FP | TP | TN | FN | FP |
| Valeurs glissantes | 46 | 686 | 12 | 0 | 58 | 686 | 0 | 0 |

TABLE 4 – Résultats de l'adaptation du profil

| Nombre de capteurs | Mois 1 t/sit (ms) | Mois 2 t/sit (ms) |
|--------------------|-------------------|-------------------|
| 20 | 2.7 | 1.7 |
| 40 | 4.8 | 3.5 |
| 100 | 22 | 20 |
| 200 | 48 | 45 |
| 400 | 102 | 100 |
| 800 | 185 | 180 |

TABLE 5 – Résultats du passage à l'échelle

nées, qui a un coût fixe par capteur.

5 Conclusions et Perspectives

Concernant la gestion des données liées à l'énergie dans un bâtiment intelligent, plusieurs types d'anomalies doivent être détectées. Elles peuvent être ponctuelles, collectives ou contextuelles. La détection de ces différents types d'anomalies est absente des méthodes de l'état de l'art pour lesquelles le prétraitement des données est obligatoire. Nous avons donc proposé SANDMAN, un système de détection d'anomalies en temps réel semi-supervisé qui utilise des données brutes comme entrée et classe les anomalies en apprenant avec les feedback de l'expert. SANDMAN est capable de détecter les trois types d'anomalies de manière générique et s'adapte bien à un nombre croissant de capteurs. Le feedback de l'expert est réduit au minimum car celui-ci est facultatif et SANDMAN peut fonctionner sans étiquetage préalable. Après réparation d'une anomalie détectée par SANDMAN, il doit automatiquement observer le retour à des valeurs nominales des capteurs pour inhiber l'anomalie en cours. C'est l'amélioration que nous développons actuellement. SANDMAN devra également être capable d'apprendre plusieurs séries de profils pour les capteurs, afin de tenir compte des différents comportements des utilisateurs des bâtiments (par exemple les périodes semaine et week-end).

Références

[1] "Energy performance of buildings." <https://ec.europa.eu/energy/en/topics/energy-efficiency/energy-performance-of-buildings>. Accessed : 2019-11-7.

[2] J. I. Guerrero, A. García, E. Personal, J. Luque, and C. León, "Heterogeneous data source integration for smart grid ecosystems based on metadata mining," *Expert Systems with Applications*, vol. 79, pp. 254–268, 2017.

[3] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection : A Survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[4] T. Labeodan, K. Aduda, G. Boxem, and W. Zeiler, "On the application of multi-agent systems in build-

ings for improved building operations, performance and smart grid interaction – A survey," *Renewable and Sustainable Energy Reviews*, vol. 50, 2015.

[5] W. Turner, A. Staino, and B. Basu, "Residential HVAC fault detection using a system identification approach," *Energy and Buildings*, vol. 151, pp. 1–17, 2017.

[6] C. Miller, Z. Nagy, and A. Schlueter, "A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings," *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 1365–1377, 2018.

[7] K. Yan, Z. Ji, and W. Shen, "Online fault detection methods for chillers combining extended kalman filter and recursive one-class SVM," *Neurocomputing*, vol. 228, pp. 205–212, 2017.

[8] J.-S. Chou and A. S. Telaga, "Real-time detection of anomalous power consumption," *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 400–411, 2014.

[9] J. Wu, W. Zeng, and F. Yan, "Hierarchical Temporal Memory method for time-series-based anomaly detection," *Neurocomputing*, vol. 273, pp. 535–546, 2018.

[10] Y. Zhu, X. Jin, and Z. Du, "Fault diagnosis for sensors in air handling unit based on neural network pre-processed by wavelet and fractal," *Energy and Buildings*, vol. 44, pp. 7–16, jan 2012.

[11] A. Capozzoli, F. Lauro, and I. Khan, "Fault detection analysis using data mining techniques for a cluster of smart office buildings," *Expert Systems with Applications*, vol. 42, no. 9, pp. 4324–4338, 2015.

[12] M. Peña, F. Biscarri, J. I. Guerrero, I. Monedero, and C. León, "Rule-based system to detect energy efficiency anomalies in smart buildings, a data mining approach," *Expert Systems with Applications*, vol. 56, pp. 242–255, 2016.

[13] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, and J. Liu, "Fault detection and operation optimization in district heating substations based on data mining techniques," *Applied Energy*, vol. 205, pp. 926–940, 2017.

[14] A. Forestiero, "Self-organizing anomaly detection in data streams," *Information Sciences*, vol. 373, pp. 321–336, 2016.

[15] Y. Seng Ng and R. Srinivasan, "Multi-agent based collaborative fault detection and identification in chemical processes," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 6, pp. 934–949, 2010.

[16] J.-P. Georgé, M.-P. Gleizes, and V. Camps, "Cooperation," in *Self-organising Software* (G. Di Marzo Seruendo, M.-P. Gleizes, and A. Karageorgos, eds.), Natural Computing Series, pp. 193–226, Springer, 2011.

[17] "Tsimulus." <https://tsimulus.readthedocs.io/en/latest/>. Accessed : 2019-12-18.

Tour d'horizon autour de l'explicabilité des modèles profonds

Gaëlle Jouis^{1,2}, Harold Mouchère¹, Fabien Picarougne¹, Alexandre Hardouin²

¹ LS2N, Université de Nantes, CNRS, F-44000 Nantes

² Pôle Emploi, Direction des Systèmes d'Information, Nantes

2-3 Juillet 2020

Résumé

Les algorithmes d'apprentissage profond ont permis d'améliorer significativement les résultats obtenus dans de nombreux domaines. Cependant, leur opacité de fonctionnement rend généralement difficile l'explication du raisonnement conduisant aux résultats. Ces modèles souffrent d'une faible acceptabilité et peuvent dans certains domaines poser un problème d'ordre légal. Pour y remédier, le domaine de l'Intelligence Artificielle eXplicable (XAI) connaît aujourd'hui un essor important. Cette contribution présente un état de l'art de différentes approches afin de rendre ces modèles plus explicables.

Mots-clés

Apprentissage Automatique, Apprentissage Profond, Explicabilité, CNN, LSTM, Réseaux à Attention

Abstract

Deep learning algorithms have significantly improved the results obtained in several fields. However, explaining the mechanisms that lead to those results can be tricky due to the operating opacity of these models. This lack of transparency leads to poor acceptance, and even to legal issues for some areas. To tackle this, the eXplainable Artificial Intelligence (XAI) research field is experiencing significant growth. This contribution introduces several state-of-the-art approaches that allow further explainability of deep models.

Keywords

Machine Learning, Deep Learning, Explainability, CNN, LSTM, Attention Networks

1 Introduction

Le domaine de l'intelligence artificielle et plus généralement de l'apprentissage a su trouver sa place dans de très nombreux secteurs de l'informatique. Depuis une dizaine d'années et dans de nombreux domaines incluant notamment le traitement d'images, la détection d'objets, le traitement textuel et linguistique, les algorithmes d'apprentissage profond ont montré leur efficacité avec des performances parfois impressionnantes [30]. À la différence d'autres approches utilisées en apprentissage, comme les modèles linéaires ou les arbres de décision par exemple, leur fonctionnement est généralement opaque et considéré

comme une *boîte noire*. Par conséquent, l'explication du raisonnement conduisant aux résultats est rendu plus difficile.

Or une plus grande transparence peut permettre d'améliorer un modèle [22, 27]. De plus, la compréhensibilité d'un modèle par un humain est intéressante dans plusieurs situations, notamment si l'on tient compte des problématiques d'acceptabilité par les utilisateurs. Cet effet *boîte noire* peut en effet engendrer des effets pervers dans l'utilisation des outils. Ainsi, si les utilisateurs n'ont pas confiance, peu importe la valeur de l'outil, il ne sera jamais exploité. A contrario, si les utilisateurs ont une confiance totale dans le système, les éventuelles dérives de l'outil ne seront pas détectées par les utilisateurs [5]. Par ailleurs, des problématiques éthiques et légales se posent lorsque les algorithmes peuvent avoir une incidence sur la vie d'une personne [13]. C'est particulièrement le cas dans le domaine médical, mais également dans l'accompagnement professionnel des personnes. Par exemple, un service public comme Pôle Emploi a l'obligation légale de pouvoir fournir les règles et algorithmes ayant une influence sur les utilisateurs ; c'est le cas si un algorithme permet de proposer des emplois aux demandeurs d'emploi. La question se pose alors de trouver le bon compromis entre la performance du modèle et son explicabilité.

La problématique de l'interprétabilité et l'acceptabilité des modèles est récente mais prend de l'ampleur. La littérature scientifique à ce sujet foisonne depuis quelques années, et des ateliers spécialisés apparaissent dans les conférences sur l'intelligence artificielle (KDD, AAI, NIPS). L'objectif de cette contribution est de faire un tour d'horizon qui ne saurait être exhaustif de différentes méthodes d'explication de modèles, en se concentrant, sans s'y contraindre, sur les techniques liées à l'analyse de textes. Dans un premier temps, un point sémantique sera fait sur la définition d'une explication, afin d'éviter toute confusion. Ensuite, une taxonomie des méthodes d'explication sera décrite, et chaque approche sera étudiée. Enfin, une discussion conclura le document.

2 Qu'est-ce qu'une explication ?

Quelques définitions. L'explicabilité est la capacité d'un humain à comprendre les décisions d'un système étant donné le contexte ; le terme "interprétabilité" est également employé [17]. Pour être explicable, un modèle doit être

compréhensible dans sa globalité, ou fournir une explication (locale) liée à chaque décision [20]. L'explication peut prendre la forme d'une justification – comme une règle logique – pour un ensemble pertinent de résultats [6]. C'est notamment le cas des Ancres [21] ou des visualisations du mécanisme d'attention [14], qui seront détaillés en sections 3.1 et 3.3 respectivement. Une explication peut également se faire par contraste : en se concentrant sur les variations menant à des décisions algorithmiques différentes. C'est le cas des vecteurs d'explication locale [2], qui indiquent quelles variations des valeurs des variables en entrée changent le résultat du modèle.

Différentes visualisations d'une explication sont présentées dans la Figure 1 : LIME (Fig. 1a), affichant l'importance de chaque mot observé pour une classe donnée, et les vecteurs d'explication locale (Fig. 1b) sont assez techniques. Les explications mettant en avant les parties importantes des images comme les ancres (Fig. 1c) et la méthode grad-CAM (Fig. 1d) sont plus visuelles. Parmi les explications illustrées, les vecteurs d'explication locale sont les seuls à fournir une vision globale du modèle.

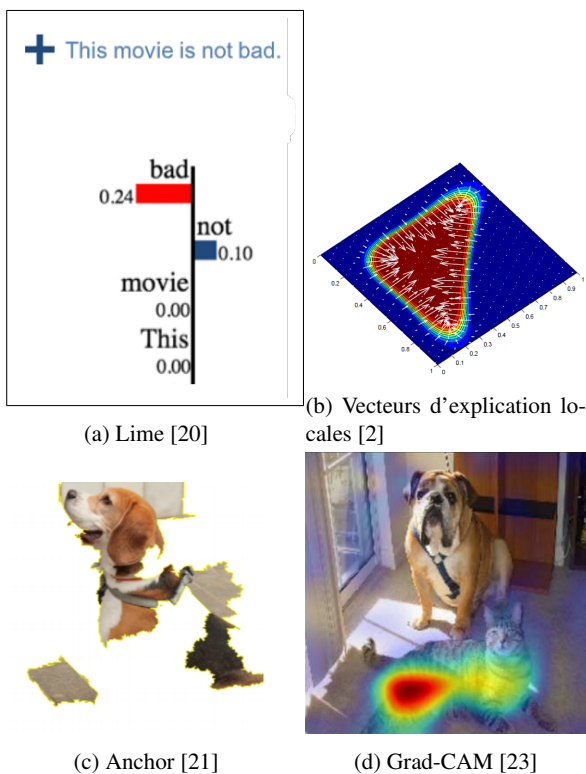


FIGURE 1 – Différentes visualisations d'une explication.

Comme le montre la figure 1, les explications peuvent avoir des formes variées. Dans un contexte donné, elles doivent être le bon compromis entre la fidélité au modèle et la compréhension pour l'utilisateur. Une explication doit alors prendre en compte l'utilisateur humain, notamment son expertise, et le temps qu'il possède pour recevoir l'explication [9].

Evaluation. Évaluer l'explicabilité (ou inversement la complexité) d'un modèle n'est pas systématiquement réalisé dans la littérature. Une métrique naïve comme la taille du modèle (nombre de paramètres) est une première approche, mais cette mesure ne rend pas compte de ce qu'a appris le modèle.

Pour plus de précision, les évaluations des explications sont réalisées avec des utilisateurs réels ou simulés (modèles appris) [20, 21]. Il est recommandé que la tâche effectuée lors de l'évaluation représente au mieux l'environnement fonctionnel dans lequel est utilisé le système créé, avec les utilisateurs réels [5]. Ceci permet de prendre en compte le contexte inhérent aux explications et les spécificités des utilisateurs.

Les métriques d'évaluation peuvent être objectives (efficacité de l'utilisateur utilisant le modèle, capacité à prédire les résultats du modèle, temps de réponse lorsque l'utilisateur a pour tâche d'imiter le modèle [21] par exemple) ou subjective quand le but est de favoriser l'acceptation d'un modèle par un utilisateur.

3 Différentes méthodes d'explication

Dans la littérature les auteurs divisent les types d'explication de modèles en fonction de la nature des approches utilisés, mais avec une même logique générale [5, 8, 9, 22]. On peut classer ces méthodes par la transparence du système étudié, ce qui donne les catégories suivantes.

1. Expliquer un modèle boîte noire au travers de ses entrées et sorties, soit en observant directement l'influence des premières sur les secondes, soit en créant un modèle interprétable mimant la boîte noire. Ces méthodes sont totalement indépendantes de la structure interne du modèle boîte noire.
2. Observer les mécanismes internes d'un système (boîte grise) après son entraînement; afin d'y détecter des schémas et les interpréter, ces méthodes sont donc dépendantes de l'architecture interne du modèle observé.
3. Concevoir un modèle ou une solution transparente (boîte blanche) de par son architecture; en lui associant des contraintes compréhensibles pour un humain (sous forme de règles par exemple) ou en générant des explications en plus du résultat attendu.

Le choix d'une méthode par rapport à une autre est généralement guidé par des contraintes techniques ou d'organisation du projet (nature du modèle, apparition de la problématique d'explicabilité avant ou après conception du modèle). Les contraintes de l'utilisateur doivent également être prises en compte : le temps à disposition pour recevoir les explications, le niveau d'expertise applicative, le niveau de connaissances en apprentissage automatique et l'intérêt porté au domaine, sont autant de paramètres qui peuvent fortement influencer sur le choix d'une méthode. De même, les explications peuvent porter sur le comportement global du modèle, ou sur un résultat particulier. Dans le second cas, on parlera dans la littérature d'explications locales [20]. Celles-ci sont particulièrement adaptées

si l'utilisateur a peu de temps, car elles se rapportent à des exemples concrets

En accord avec la taxonomie présentée ci-avant, chacune des trois approches sera détaillée au travers de plusieurs méthodes.

3.1 Explications indépendantes du modèle

Ce type de méthode a pour principal avantage de pouvoir être utilisé sur tous les modèles. Cela évite de se contraindre techniquement dans le choix d'un modèle. Les explications sont basées sur l'impact des valeurs des variables d'entrée sur la sortie du modèle. Les explications relèvent donc de la corrélation, pas nécessairement de la causalité. De même, les variables en entrée peuvent être trop nombreuses pour être assimilées aisément par un utilisateur. C'est par exemple le cas en analyse sémantique, où ces entrées sont des n-grammes de mots ou de caractères, notion illustrée ci dessous (Fig. 2).

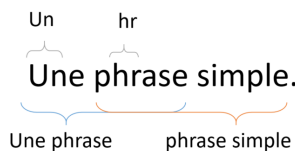


FIGURE 2 – Exemples de 2-grammes de caractères (en haut) et de mots (en bas).

Vecteurs d'explication locale. Dans [2] les auteurs présentent des *vecteurs d'explication locale* qui quantifient l'importance de chaque variable d'entrée pour une instance donnée. Plus exactement, un vecteur d'explication indique dans quelle direction changer une variable pour que le modèle change de classe prédite. On obtient donc un vecteur indiquant le rôle de chaque variable dans la classification d'une instance, pour une classe donnée. En observant un ensemble de vecteurs, on peut également avoir une visualisation plus globale du modèle. Une illustration est donnée en Fig. 1b avec deux classes (bleu ou rouge) et deux variables d'entrées correspondant aux deux dimensions du plan. Le calcul du vecteur d'explication se fait sur des modèles donnant une probabilité pour une entrée d'appartenir à chaque classe. Si elle n'est pas disponible, il faut l'estimer avec un autre modèle avant de calculer les vecteurs. C'est donc une approche qui n'est pas totalement indépendante du modèle. Par ailleurs, ces vecteurs sont difficilement interprétables pour les utilisateurs, surtout dans le cas de l'analyse sémantique où la dimension est la taille du vocabulaire du corpus. Il faut donc envisager une visualisation ou un résumé de l'information apportée par ces vecteurs pour en tirer une explication.

Valeurs de Shapley. Inspiré de la théorie des jeux, les valeurs de Shapley donnent un aperçu de la contribution d'un élément dans un ensemble par rapport à un résultat final [26]. La dimension des explications obtenues est égale à la dimension des données en entrée. Elles nécessitent donc un traitement a posteriori pour en retirer des explications claires. Le calcul des valeurs de Shapley né-

cessite d'avoir accès au jeu de données d'entraînement du modèle, ce qui est une limitation forte et sous-entend un calcul éventuellement long. Pour limiter le temps de calcul, le module SHAP (SHapley Additive exPlanation) a été développé en se basant sur les valeurs de Shapley [16]. Toutefois, les méthodes proposées sont des estimations des valeurs SHAP, avec plusieurs approches présentées, dépendantes et indépendantes du modèle étudié.

Distillation de connaissance. Une autre manière de rendre les modèles interprétables est de se concentrer sur des structures plus petites. Pour cela, les auteurs de [11] présentent le concept de la distillation de connaissances dans les modèles, où l'idée est d'entraîner un ensemble complexe de réseaux de neurones et transférer les structures apprises à un réseau de neurones plus simple. Instinctivement, il est tentant de se dire qu'un réseau plus simple pourrait être plus interprétable. Les auteurs de [7] proposent de mesurer l'interprétabilité comme étant le ratio entre la performance du modèle simple et du modèle complexe. En appliquant ces mesures sur un cas d'étude avec un réseau de neurones complexe et un réseau de neurones simplifié, ils en déduisent que la Distillation améliore la robustesse (résistance aux attaques par modifications subtiles des entrées du modèle) au détriment de l'interprétabilité. En revanche, la Distillation est également utilisée pour créer des arbres de décision [15] ou des Gradient Boosted Trees [3] à partir de réseaux de neurones, ce qui revient à créer un module d'explication sous forme d'arbres de décisions. Si l'approche est décrite pour des réseaux de neurones, elle est applicable à d'autres modèles.

Approximation linéaire locale. Les auteurs de [20] proposent l'outil *LIME* (Local Interpretable Model-agnostic Explanations) pour rendre compte du comportement local d'un modèle. *LIME* fonctionne par approximation linéaire du modèle autour d'une instance donnée. C'est ce modèle linéaire qui est ensuite utilisé pour générer des explications. Ces explications correspondent aux variables d'entrée qui impactent le plus la sortie du modèle (Fig. 1a). Pour l'analyse de texte, ce sont les mots du texte associés à une quantification de l'influence (positive ou négative) sur la réponse du modèle.

Pour obtenir une explication plus globale d'un modèle, SP-LIME est présenté en [20]. SP-LIME est une méthode se basant sur les explications locales de *LIME* pour en sélectionner un ensemble restreint. Cette méthode est intéressante car elle est généralisable au-delà de *LIME* [21]. Par contre, elle nécessite de générer en amont toutes les explications des instances d'un ensemble de données pour pouvoir les sélectionner ensuite. Cet ensemble de données étant idéalement un ensemble de test.

Les avantages de cette approximation linéaire sont la simplicité et la rapidité de la création de l'explication. Toutefois il faut faire l'hypothèse forte que le modèle se comporte linéairement autour de l'instance expliquée. Le principal inconvénient est le fait que l'explication soit une approximation locale du modèle sans limite définie. Il est en effet difficile pour un utilisateur humain de savoir à quel

point ces explications sont généralisables.

Ancres. Pour contrer ce problème, les auteurs de LIME ont proposé une amélioration de leur méthode, les Ancres [21]. En conservant l'idée d'approximation locale du modèle, les auteurs sont passés d'une approximation par un modèle linéaire à une explication sous forme de règle. L'idée est de mieux définir le contexte dans lequel l'explication générée est valable. Soient un modèle $f : X \rightarrow Y$, une instance $x \in X$, un résultat $y \in Y$ choisi, et une ancre A associée. A est une condition telle que si x respecte cette condition, alors la probabilité que $f(x) = y$ est grande. L'ancre est construite de sorte à maximiser cette probabilité, il est toutefois possible que $f(x) \neq y$. On note $D(\cdot|A)$ l'ensemble des $x \in X$ qui respectent la condition A . Une ancre intéressante s'applique à un ensemble $D(\cdot|A)$ le plus grand possible relativement à X . Si les entrées sont des textes, une ancre est un ensemble de mots ou n-grammes.

Dans l'exemple illustré en Fig. 3, le modèle étudié classe des phrases selon deux catégories : "positive" et "négative". L'instance d'origine est la phrase "This movie is not bad", et est classée "positive". L'ancre associée est la règle $A = \{not, bad\} \rightarrow Positive$. L'ensemble $D(\cdot|A)$ de la Fig. 3a, représenté par un rectangle dans la Fig. 3b, regroupe les entrées possédant les variables de l'ancre. Dans notre exemple, $D(\cdot|A)$ correspond aux textes comprenant les mots "not" et "bad" de l'ensemble des variations de l'instance d'origine (ensemble D de la Fig. 3). L'ensemble D est obtenu en appliquant des variations cohérentes à l'instance d'origine. Appliqué à l'exemple, cela correspond à remplacer un ou plusieurs mots de la phrase par des mots de nature similaire. Remplacer un adjectif par un autre adjectif est une variation cohérente de l'instance d'origine.

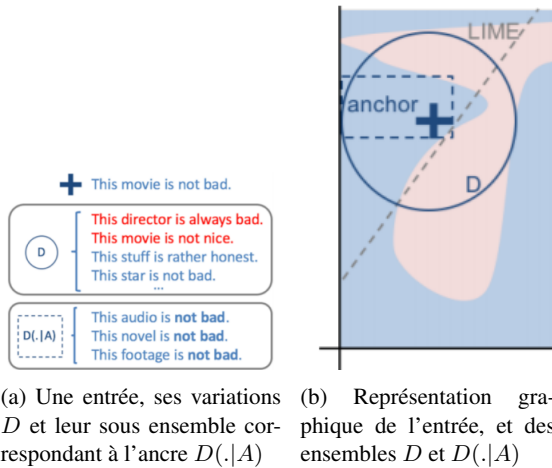


FIGURE 3 – Textes similaires à une entrée D et son sous ensemble $D(\cdot|A)$ correspondant à l'ancre $\{not, bad\} \rightarrow Positive$. Source : [21]

Pour sélectionner une ancre, les auteurs choisissent de maximiser deux paramètres. Le premier est la précision, qui est maximale si les éléments de l'ensemble $D(\cdot|A)$ ont la même sortie que l'instance d'origine. Le second est la couverture, soit la taille de l'ensemble $D(\cdot|A)$ par rapport

à l'ensemble D . Une ancre avec une forte précision est une explication fidèle au modèle boîte noire. Si elle a une bonne couverture, une ancre est assez généralisable.

Le travail autour des ancres met en avant la relation entre l'utilisateur et l'explication. La limite de LIME contournée par les Ancres est la difficulté pour l'utilisateur à déterminer la validité d'une l'explication donnée. Ce faisant, l'explication donnée, à savoir une règle, est également plus facile à aborder qu'un ensemble de poids comme le résultat de base de LIME.

Les explications indépendantes des modèles sont une première approche en considérant les modèles boîtes noires, mais il s'agit plus de mettre en avant des corrélations entre les entrées et les résultats du système. Pour aller plus loin, il est possible de s'appuyer sur la structure du modèle pour générer des explications. La *boîte noire* devient alors une *boîte grise*.

3.2 Explications dépendantes du modèle

Les explications dépendantes du modèle sont basées sur l'observation des paramètres du modèle après son entraînement. Cette observation, pour conserver la métaphore de la boîte noire, revient à ouvrir cette boîte et regarder à l'intérieur. Les explications sont alors plus fidèles au fonctionnement du système étudié que les méthodes indépendantes du modèle. Toutefois, cette approche contraint fortement le choix du modèle.

Réseaux à convolutions. Dès 2013, des travaux sur les réseaux de neurones proposent des visualisations de leur fonctionnement, dans le cadre de la classification d'images. Dans [29], les auteurs proposent de mieux appréhender le fonctionnement des réseaux à convolution (Convolutional Neural Networks, CNN), par une visualisation directe des motifs d'activation des neurones par couche. Ils utilisent pour cela un réseau de neurones appelé *Deconvolutional Network*. Ces travaux permettent d'avoir un aperçu des motifs reconnus par chaque couche, motifs simples sur les couches basses et plus semblables aux classes détectées sur les couches hautes. Les auteurs vérifient également le comportement de leur modèle en masquant certaines parties des images et en observant les éventuelles variations de classification induites; cette approche étant reprise dans [20] et [21].

Toujours sur les CNN, les auteurs de [23] présentent la méthode *Grad-CAM*. Elle permet de générer des cartes de chaleur des endroits de l'image aidant à la détection d'une classe en particulier (Fig 1d). Ils mélangent leurs visualisations à celles de [25, 29], afin d'obtenir les parties de l'image ainsi que les motifs précis permettant la classification. Des travaux sont également effectués sur les cartographies des caractéristiques saillantes (*Saliency maps*) des images [24]. Ces travaux permettent d'avoir une indication fidèle du fonctionnement du modèle, mais les techniques sont plutôt orientées analyse d'image.

Long-Short Term Memory (LSTM). Dans la même philosophie, les auteurs de [12] essaient de comprendre les forces et les limites des réseaux de neurones de type

LSTM, appliqués à l'analyse de textes. Pour leur analyse, ils génèrent des visualisations sur des motifs spécifiques dans les données entraînant les activations de certaines cellules. Un exemple du papier est l'activation de certaines cellules en fonction des caractères rencontrés dans un texte. La visualisation met en évidence la détection du texte entre guillemets.

Cell that turns on inside quotes:
 "You mean to imply that I have nothing to eat out of... On the contrary, I can supply you with everything even if you want to give dinner parties." warmly replied Chichagov, who tried by every word to prove his own rectitude and therefore imagined Kutuzov to be animated by the same desire.
 Kutuzov, shrugging his shoulders, replied with his subtle penetrating smile: "I meant merely to say what I said."

FIGURE 4 – Activation d'une cellule en fonction des guillemets dans le texte. Source : [12]

Ce type de réseau de neurones est largement utilisé dans l'analyse sémantique. Les visualisations peuvent donner une idée précise du fonctionnement du modèle. toutefois, elles sont complexes ou nécessitent un travail de recherche et d'analyse, d'autant plus si on considère chaque cellule. Pour donner un ordre d'idée, des structures de LSTM de la littérature peuvent avoir 300 [14] ou encore 512 [12] cellules. Il convient alors de toutes les explorer pour trouver, pour une petite partie d'entre elles, des activations significatives. Si ce type d'approche permet de mieux comprendre les indices utilisés par le réseau, il ne s'agit pas forcément d'une explication de la décision finale.

Décomposition pixel par pixel. La décomposition pixel par pixel (*Pixel-Wise Decomposition*) est une stratégie permettant d'expliquer les résultats d'un classifieur d'images en créant une carte de chaleur des pixels les plus pertinents pour une prédiction donnée [1]. Pour calculer la pertinence R (*Relevance*) de leurs variables d'entrée (dans leur exemple, des pixels), ils décomposent la prédiction $f(x)$ comme étant la somme des contributions des neurones de la couche précédente et appliquent itérativement cette propriété jusqu'à arriver à la couche d'entrée de leur réseau. Les auteurs décrivent deux manières d'y parvenir. La première est la propagation de pertinence couche par couche (*Layer-wise Relevance Propagation*), un concept regroupant diverses solutions de décomposition respectant certains critères. La seconde est une approche basée sur la décomposition de Taylor, qui permet une approximation de la propagation de pertinence couche par couche, en s'appuyant sur le principe de décomposition de fonctions pour décomposer directement le classifieur f . Ces travaux sont approfondis dans [18] où les auteurs proposent la *Deep Taylor Decomposition*, qui est une adaptation de la décomposition de Taylor, appliquée non pas au modèle entier mais à chaque fonction de pertinence $R_j(x_i)$ entre un neurone j d'une couche n et les neurones x_i de la couche $n - 1$. Dans ce dernier article les auteurs illustrent leur travaux avec des réseaux de neurones classant des images, mais ce type de méthode peut être appliqué à d'autres modèles et peut également être élargi à d'autres types d'entrées, comme les textes dans [18].

Les méthodes décrites précédemment permettent de mieux

appréhender le fonctionnement des modèles, en se basant sur leurs caractéristiques respectives. Toutefois ces méthodes, dépendantes ou indépendantes du modèle, nécessitent des calculs ou de l'analyse d'un grand nombre d'éléments. Pour éviter cela, la dernière approche est de créer un modèle dont la structure même le rend plus transparent.

3.3 Modèle Interprétable

L'idée de modèle transparent est de limiter le besoin en analyse post entraînement en s'appuyant sur des structures spécifiques du modèle. De cette manière une meilleure fidélité au modèle est assurée tout en limitant les calculs et approximations.

Architectures simplifiées. Dans le courant de simplification des réseaux, des expérimentations mettent en évidence l'efficacité d'architectures de réseaux de neurones simplistes mais capables de résoudre des problèmes complexes, comme le stationnement d'une voiture miniature [10]. Ce type de réseau possède une topologie inspirée du système nerveux du ver *C. elegans*. Le papier définit un réseau ainsi constitué de 12 neurones, en l'entraînant sur la tâche de stationner un robot. En observant les activations des neurones en fonction des phases (tourner à gauche, à droite ou avancer), les auteurs mettent en lumière le rôle des neurones dans chaque phase de la tâche accomplie, en mettant en évidence par exemple les neurones s'activant lorsque le robot doit tourner à droite. Ces activations sont interprétables notamment parce que le réseau est composé de peu de neurones; cela facilitant grandement l'analyse des activations. Cette approche permet ainsi de réaliser un travail similaire à [12], qui analyse les activations de cellules LSTM, mais sur un nombre réduit de neurones.

Mécanismes d'attention. Les mécanismes d'attention dans les réseaux de neurones sont une manière de rendre les modèles directement plus interprétables. Dans [14], les auteurs créent un plongement de mots via un réseau avec une partie LSTM et une partie basée sur l'attention. Chaque phrase est représentée par une matrice $M = AH$, où A est la matrice d'attention et H les états cachés de la couche LSTM. Les vecteurs d'attention a composant A vont se concentrer sur des aspects différents de la phrase. En sommant et normalisant (softmax) tous les vecteurs d'attention a , les mots fortement considérés par le plongement ressortent avec les poids les plus forts. Cette solution permet donc d'avoir une visualisation claire des variables importantes en entrée du réseau, en observant les paramètres du modèle. Un exemple de visualisation est présenté dans le cadre de la traduction de textes dans [19] (Fig. 5). Elle met en avant l'inversion des mots entre l'entrée en anglais (*European economic area*) et la traduction française (*zone économique européenne*). La visualisation des poids d'attention est également utilisée dans [28] afin de valider l'intérêt de leur topologie de réseau, également composé d'un LSTM et d'une couche d'attention. Le principal intérêt de l'analyse de l'attention est qu'il n'y a pas besoin de calculs supplémentaires d'une métrique spécifique une fois le modèle entraîné, contrairement à [1] par exemple.

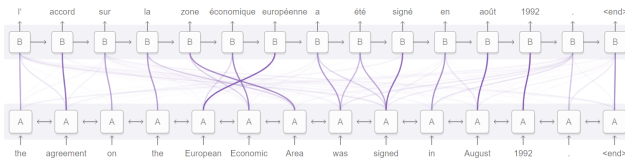


FIGURE 5 – Visualisation de l'attention pour une tâche de traduction. Source : [19]

Génération d'explications. Certains systèmes peuvent également générer d'eux même des explications autour d'une décision. C'est le cas de [4] où les explications d'un système de recommandation sont générées sous forme de critiques d'utilisateurs ("i wouldn't recommend it.") via un LSTM. Le principe de l'expérimentation est de reconstruire une critique que produirait un utilisateur avec un texte le plus naturel possible. Les auteurs évaluent leurs explications avec des métriques de lisibilité de textes tel que le score *Flesch-Reading-Ease*. Si le système de recommandation n'est pas nécessairement un réseau de neurones, un système similaire peut être appliqué à tout système de prédiction basé sur des textes.

Ce tour d'horizon de différentes méthodes donne un aperçu des approches possibles, mettant en avant la fidélité de l'explication donnée ou sa capacité à être comprise et acceptée par l'utilisateur final.

4 Discussion

L'analyse de la littérature met en lumière un nombre significatif de méthodes, parfois très proches. Leur finalité commune est d'améliorer l'explicabilité des modèles, en particulier dans le domaine de l'apprentissage profond. Pour chaque problème, il s'agit alors de trouver et d'adapter les méthodes en fonction des contraintes rencontrées. Le choix d'une approche d'explication peut alors devenir une contrainte technique du modèle et guider les développements futurs. Ainsi plus la réflexion est menée tôt, plus le choix des approches est grand.

Ces contraintes sont aussi humaines : le public cible des explications doit être défini et caractérisé. Les notions d'intérêt et de confiance dans les technologies dites d'intelligence artificielle sont des caractéristiques importantes des utilisateurs. La prise en compte du temps disponible pour recevoir l'explication est également primordiale.

Un enjeu important du travail autour de l'explicabilité est de normaliser l'évaluation des différentes méthodes et de leurs adaptations. Aujourd'hui, ces évaluations ne sont pas systématisées et il n'existe pas de consensus, rendant la comparaison des méthodes délicate.

Références

[1] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations

for non-linear classifier decisions by layer-wise relevance propagation. *PLoS one*, 10(7) :e0130140, 2015.

- [2] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *J. Mach. Learn. Res.*, 11 :1803–1831, August 2010.
- [3] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Distilling knowledge from deep networks with applications to healthcare domain. *arXiv preprint arXiv :1512.03542*, 2015.
- [4] Felipe Costa, Sixun Ouyang, Peter Dolog, and Aonghus Lawlor. Automatic generation of natural language explanations. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, pages 1–2, 2018.
- [5] Hoa Khanh Dam, Truyen Tran, and Aditya Ghose. Explainable software analytics. In *Proceedings of the 40th International Conference on Software Engineering : New Ideas and Emerging Results*, pages 53–56, 2018.
- [6] Sonia Desmoulin-Canselier and Daniel Le Métayer. Algorithmic decision systems in the health and justice sectors : Certification and explanations for algorithms in european and french law. *European Journal of Law and Technology*, 9(3), 2019.
- [7] Amit Dhurandhar, Vijay Iyengar, Ronny Luss, and Karthikeyan Shanmugam. Tip : Typifying the interpretability of procedures. *arXiv preprint arXiv :1706.02952*, 2017.
- [8] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations : An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE, 2018.
- [9] Riccardo Guidotti, Anna Monreale, Salvatore Ruggeri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5) :1–42, 2018.
- [10] Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Can a compact neuronal circuit policy be re-purposed to learn simple robotic control ?
- [11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv :1503.02531*, 2015.
- [12] Andrej Karpathy, Justin Johnson, and Fei-Fei Li. Visualizing and understanding recurrent networks. *CoRR*, abs/1506.02078, 2015.
- [13] Legifrance. Loi n 2016-1321 du 7 octobre 2016 pour une republique numerique, 2016.

- [14] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. *arXiv preprint arXiv :1703.03130*, 2017.
- [15] Xuan Liu, Xiaoguang Wang, and Stan Matwin. Improving the interpretability of deep neural networks with knowledge distillation. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 905–912. IEEE, 2018.
- [16] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.
- [17] Tim Miller. Explanation in artificial intelligence : Insights from the social sciences. *Artificial Intelligence*, 267 :1–38, 2019.
- [18] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65 :211–222, 2017.
- [19] Chris Olah and Shan Carter. Attention and augmented recurrent neural networks. *Distill*, 2016.
- [20] Marco Tulio Ribeiro, Sameer Singh, and Carlos 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, KDD '16*, pages 1135–1144, New York, NY, USA, 2016. ACM.
- [21] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors : High-precision model-agnostic explanations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1527–1535, 2018.
- [22] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence : Understanding, visualizing and interpreting deep learning models. *ITU Journal : ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services*, 1 :1–10, 10 2017.
- [23] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam : Visual explanations from deep networks via gradient-based localization. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [24] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks : Visualizing image classification models and saliency maps. *arXiv preprint arXiv :1312.6034*, 2013.
- [25] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity : The all convolutional net. *arXiv preprint arXiv :1412.6806*, 2014.
- [26] Erik Strumbelj and Igor Kononenko. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 11(Jan) :1–18, 2010.
- [27] Stefano Teso and Kristian Kersting. Explanatory interactive machine learning. In [http ://www. aies-conference. com/accepted-papers/](http://www.aies-conference.com/accepted-papers/). AAAI, 2019.
- [28] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615, 2016.
- [29] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [30] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years : A survey. *arXiv preprint arXiv :1905.05055*, 2019.

Apprentissage profond appliqué à la classification d'images microscopiques embryonnaires

T. Gomez¹, H. Mouchère¹, T. Fréour^{2,3}, M. Feyeux⁴,

¹ LS2N, CNRS, Université de Nantes, France

² Service de médecine et biologie du développement et de la reproduction, CHU Nantes, France

³ CRTI, Inserm, Université de Nantes, Nantes, France

⁴ SFR Santé, MicroPICell core facility, CNRS, INSERM, Université de Nantes, Nantes, France

29 Juin 2020

Résumé

Dans le domaine de la fécondation in vitro (FIV), l'imagerie par time-lapse (ITL) est une technologie qui produit une vidéo montrant le développement de l'embryon pendant ses premiers jours. L'ITL permet d'annoter les instants auxquels commencent et finissent les différentes phases de développement de l'embryon. Cependant, le processus d'annotation prend du temps et nécessite des experts. Dans cet article, nous montrons que pour prédire les phases visibles dans les images issues de vidéos ITL, un ResNet-3D est meilleur qu'un équivalent 2D ou un ResNet-LSTM.

Mots-clés

apprentissage profond, fécondation in vitro, validation croisée, modèle temporel, convolution 3D, ResNet, LSTM, CNN, apprentissage machine

Abstract

In the field of in vitro fertilization (IVF), time-lapse imaging (TLI) is a technology that produces a video showing the development of the embryo during its first days. TLI makes it possible to annotate the moments at which the different stages of embryo development begin and end. However, the annotation process takes time and requires experts. In this article, we show that to predict the visible phases in images from TLI videos, a ResNet-3D is better than a 2D equivalent or a ResNet-LSTM.

Keywords

deep learning, in vitro fertilization, cross validation, time model, 3D convolution, ResNet, LSTM, CNN, machine learning

1 Introduction

L'infertilité est encore un problème mondial aujourd'hui puisqu'il touche 186 millions de personnes [1] et que le nombre de couples déclarant leur infertilité dans des centres en Europe augmente de 8 – 9% par an [2]. La fécondation in vitro (FIV) est l'un des traitements les plus courants de l'infertilité. Elle implique une stimulation ovarienne suivie du prélèvement de plusieurs ovocytes, de la

fécondation et de la culture d'embryons pendant 1 à 6 jours dans des conditions environnementales contrôlées. Les embryologistes sélectionnent ensuite un embryon à transférer en fonction de leur évaluation de la qualité des embryons. L'efficacité de la FIV n'est pas encore optimale aujourd'hui [3]. Cette situation s'explique en partie par les limites actuelles des méthodes d'évaluation de la qualité des embryons. La méthode d'évaluation la plus courante est l'évaluation morphologique qui consiste à observer l'embryon au microscope régulièrement pendant ses premiers jours et à évaluer sa morphologie. Le croissance des embryons étant très lente, les biologistes n'obtiennent qu'une vision statique du développement. Cette méthode souffre toujours d'un manque de pouvoir prédictif et d'une grande variabilité inter et intra-biologiste [4].

L'Embryoscope[®] est un appareil qui permet aux embryons de se développer dans des conditions environnementales contrôlées et qui intègre une technologie d'ITL. L'ITL est une technologie récente qui consiste à prendre plusieurs photographies de l'embryon par heure pendant quelques jours et qui produit une vidéo time-lapse donnant une vision dynamique du développement. Elle permet d'observer en continu l'embryon sans le retirer des conditions contrôlées et stables de l'incubateur. Cette technologie permet une évaluation morpho-cinétique (MC) des embryons, c'est-à-dire que les biologistes ont une vision dynamique du développement de l'embryon et peuvent annoter certains événements temporels (divisions des cellules, expansion du blastocyste, etc.).

Un travail récent a montré que la surveillance des vidéos time-lapse est associée à un taux de grossesse clinique et de naissance vivante significativement plus élevé, et à une perte précoce de grossesse significativement plus faible par rapport à la sélection morphologique [5] mais une autre étude a donné des résultats contradictoires [6]. En outre, plusieurs travaux [7, 8] ont proposé des modèles prédictifs de l'issue de la FIV (grossesse/non grossesse) basés sur des bases de données MC. Même si la variabilité a diminué grâce à l'ITL, elle reste un problème [9].

Cependant, l'annotation des paramètres MC est un processus qui prend du temps et qui nécessite des embryologistes

expérimentés. Dans cet article, nous évaluons rigoureusement plusieurs algorithmes pour prédire automatiquement les paramètres MC en utilisant des méthodes d'apprentissage profonds.

2 Description du problème

L'extraction automatique des paramètres-morphocinétique (MC) est une tâche de classification d'images qui consiste à classer chacune des images des vidéos en plusieurs classes, qui sont les différentes phases de développement auxquelles l'embryon peut se trouver. Les phases de développement étudiées dans les travaux cités ci-dessous sont choisies parmi les suivantes : $tPB2$, $tPNa$, $tPNf$, $t2$, $t3$, $t4$, $t5$, $t6$, $t7$, $t8$, $t9+$, tM , tSB , tB et enfin tEB . Le développement se déroule de la manière suivante : d'abord les pro-nucléi apparaissent puis disparaissent (phases $tPB2$, $tPNa$, $tPNf$), puis une cellule unique se divise jusqu'à qu'il y ait environ une dizaine de cellules (phases $t2$ à $t9+$), ensuite les cellules se compactent, le blastocyste apparaît et enfin ce dernier s'agrandit (phases tM , tSB , tB , tEB). Chacune de ces phases est définie en [10] et certaines sont illustrées en Figure 1. Ces phases sont ordonnées, il est très rare qu'un embryon revienne à une phase précédente durant son développement.

Chaque vidéo montre le développement d'un embryon particulier. Pour chaque vidéo, un biologiste expérimenté note l'heure de début et de fin de chaque phase de développement de l'embryon. Chaque image de chaque vidéo a donc une classe, qui est la phase à laquelle l'embryon se trouve dans cette image.

Il faut noter que certaines phases peuvent ne pas être visibles dans la vidéo. Par exemple un embryon peut contenir 2 cellules (phase $t2$) puis ces deux cellules peuvent se diviser chacune dans l'intervalle de temps qui sépare deux images consécutives (phase $t4$) sans qu'il n'y ait un moment dans la vidéo où l'embryon présente clairement 3 cellules (phase $t3$).

Chaque vidéo contient entre 300 et 600 images en nuance de gris et ont une résolution de 500×500 pixels. La durée réelle qui sépare deux images est de 10 à 20 minutes.

L'embryon est un objet en trois dimensions et il est parfois nécessaire d'ajuster le plan focal de l'Embryoscope afin de pouvoir observer correctement toutes les parties de l'embryon. Certains travaux extraient donc plusieurs vidéos par embryon, chacune enregistrée avec un plan focal différent, afin de donner une meilleure vision de l'embryon au modèle. Utiliser plusieurs plans focaux demande beaucoup de calculs, il est donc aussi possible d'extraire une seule vidéo en laissant l'Embryoscope choisir le meilleur plan focal à chaque image, c'est-à-dire le plan qui maximise la netteté de l'image.

Les différentes phases n'ont pas toutes la même durée donc les classes sont déséquilibrées. Par exemple la classe majoritaire est la classe $t9+$ et constitue 16% du total des images alors que la classe la moins fréquente ($t3$) constitue moins de 2% des images.

3 État de l'art

Nous présentons d'abord des travaux traitant de l'extraction automatique des paramètres MC et ensuite quelques travaux sur des tâches connexes : segmentation automatique de la masse cellulaire interne (MCI), du trophectoderme (TE) et enfin prédiction automatique de la qualité embryonnaire. Les approches classiques de classifications d'images et de segmentation de vidéos ont été largement utilisés.

3.1 Extractions des paramètres morphocinétiques

Plusieurs travaux ont tenté d'extraire automatiquement les informations morpho-cinétiques des vidéos en time-lapse brutes.

Certains travaux proposent une version simplifiée du problème avec moins de classes. Les auteurs de [11] et [12] se concentrent sur les phases entre 1 et 5 cellules et les auteurs de [13] proposent de ne modéliser que deux classes : blastocyste et non-blastocyste. Kanakasabapathy et al. [13] font valoir que la réduction du nombre de classes permet d'avoir des annotations plus nettes, car il y a beaucoup moins de variabilité intra et inter-opérateurs lorsque l'on ne modélise que ces deux classes. Des modèles comme AlexNet [11], un ensemble de U-Nets [14] dilatés résiduels [12] ou Xception [15, 13] sont utilisés comme classifieur. D'autres travaux [16, 17] proposent de travailler avec un nombre réduit de phases, à savoir les 6 premières phases de développement.

Le contexte obtenu en utilisant plusieurs trames voisines a également été exploité. La plupart des travaux proposent une fusion tardive en calculant les mêmes traits sur plusieurs trames successives, puis en agrégeant les informations pour produire une prédiction. Par exemple, il a été proposé de concaténer les vecteurs caractéristiques des images calculées par le même réseau de neurones convolutifs (CNN) [16], ou alors de les agréger à l'aide de max-pooling [17]. Il a aussi été montré qu'il est avantageux, lorsque plusieurs images sont utilisées, de prédire l'étiquette des images voisines en utilisant les caractéristiques de l'image actuelle [17]. Enfin, Lau et al. [18] utilisent un LSTM [19] pour prendre en compte les caractéristiques extraites à chaque pas de temps. Il faut noter que la fusion précoce d'informations a également été proposée dans [16] en concaténant les images sur la dimension du canal. Les différentes phases de développement sont strictement ordonnées : l'embryon peut parfois sauter une phase mais il est très rare qu'il revienne à une phase précédente. Cette contrainte est souvent exploitée par la programmation dynamique pour rendre la prédiction cohérente sur toute la vidéo [11, 18, 17, 16],

Plusieurs méthodes produisent des informations spatiales sur le blastocyste. Par exemple, il est possible d'apprendre à déduire la position du centroïde de chaque cellule de manière totalement supervisée [12]. Lau et al. utilisent l'apprentissage par renforcement pour recadrer l'image d'entrée sur une région d'intérêt avec un réseau de proposition

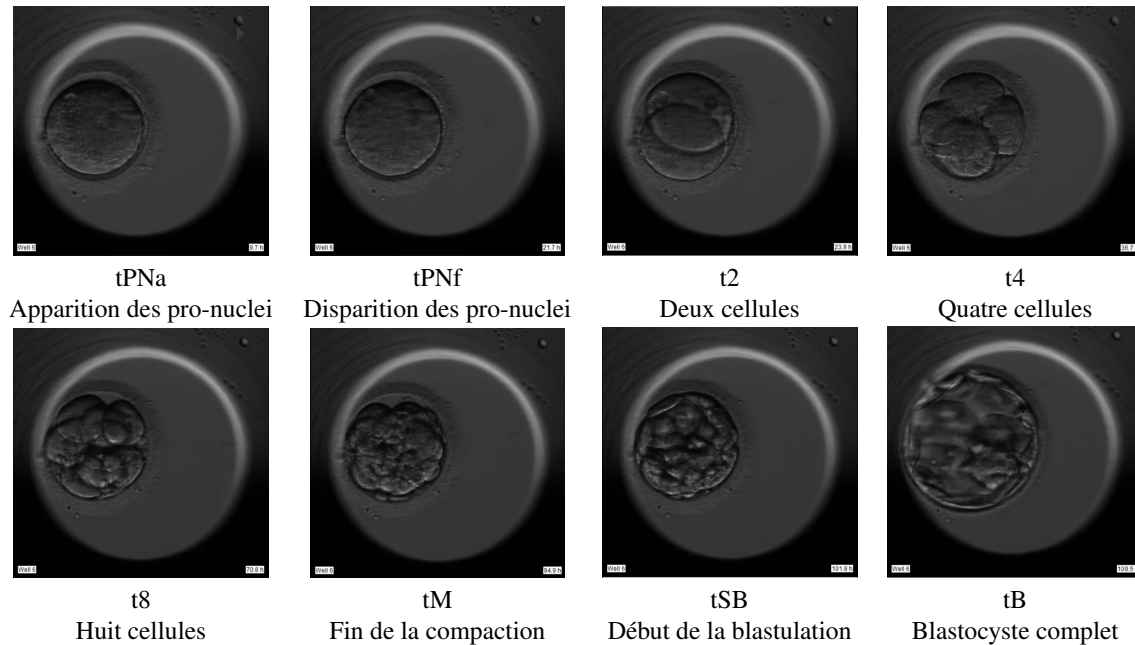


FIGURE 1 – Illustrations de 8 des 15 phases que nous utilisons

de région (RPN) [20] et à passer ensuite le résultat à un CNN [18]. L'image est divisée en une grille de 32×32 cellules et le RPN produit une distribution de probabilité sur celle-ci. Une cellule est choisie aléatoirement et la partie de l'image contenue dans la cellule est envoyée au classificateur. Ce dernier est entraîné en utilisant une entropie croisée et le RPN est entraîné en utilisant la méthode de "policy gradient" basé sur la performance du classificateur. Pour forcer le RPN à produire diverses distributions de probabilité, un terme d'entropie est ajouté à la fonction de coût. Ce détecteur apprend à localiser le blastocyste de manière faiblement supervisée.

3.2 Tâches liées

Les articles examinés ici couvrent les tâches suivantes : segmentation du TE, segmentation de la MCI et qualité des embryons. Pour extraire automatiquement la région du TE, une méthode *ad-hoc* basée sur l'algorithme Retinex a été développée [21]. Les auteurs l'utilisent pour nettoyer les régions à faible gradient, puis appliquent la méthode Canny Edge pour obtenir une estimation brute du TE. Enfin, ils utilisent l'algorithme des k-moyennes pour remplacer les composantes connexes par le centroïde auquel ils appartiennent, ce qui réduit le bruit dans le résultat. Pour segmenter la MCI, un FCN [22] et l'architecture U-Net [14] ont respectivement été proposés en [23] et [24]. Kheradmand et al. [23] utilisent un ensemble de données de 235 images augmentées de 36 rotations pour chaque image. Rad et al. [24] utilisent le même ensemble de données sans augmentation. Les auteurs ajoutent plusieurs convolutions dilatées dans la partie centrale de l'architecture afin d'augmenter le champ réceptif des neurones. Ils utilisent également un ensemble de quatre U-Nets différents, chacun ayant une résolution d'entrée différente (448×448 ,

384×384 , 320×320 , et 256×256), pour capturer un contexte plus large.

Les auteurs de [25] proposent des traits conçus à la main pour prédire la qualité des embryons bovins. Ils calculent d'abord la matrice de co-occurrence de niveau de gris (MCNG), la segmentation des blastocystes à l'aide de la méthode de transformée circulaire de Hough et la segmentation MCI à l'aide de la méthode des bassins versants. Ensuite, 36 variables scalaires sont calculées en utilisant les deux segmentations et le MCNG. Ces variables comprennent des traits comme le contraste, la corrélation, l'énergie, etc. Khosravi et al. [26] utilisent 10 000 images d'embryons, capturés à 110h post-insémination avec tous les plans focaux disponibles. Ils utilisent la méthode Veeck et Zaninovic [27] pour classer chaque embryon. Cette méthode manuelle attribue trois notes à l'embryon en fonction du développement du blastocyste, de la qualité de la MCI, et de la qualité du TE. Ces notes sont basées sur des critères morphologiques estimés par l'expert. Ensuite, ils classent les embryons en trois groupes : bonne qualité, qualité moyenne et mauvaise qualité (en fonction du taux de grossesse). Pour chaque embryon, 6000 images du groupe de bonne qualité et 6000 images du groupe de mauvaise qualité sont choisies au hasard. Un CNN [28] est entraîné pour faire la distinction entre les deux groupes. Enfin, ils étudient le lien entre la qualité de l'embryon et l'âge du patient à l'aide d'un arbre de décision.

Les auteurs de [8] utilisent les paramètres morphocinétiques, les niveaux de fragmentation, la présence de multinucléation, l'uniformité des blastomères et l'âge des femmes passés comme entrée d'un NN en essayant de prédire s'il y aura ou non une implantation. La corrélation entre les caractéristiques de l'ACP et la grossesse a égale-

ment été étudiée. Milewski et al. [29] utilisent des images de 16 000 embryons, capturés entre 112 et 116 heures (jour 5) ou 136 à 140 heures (jour 6) de l’insémination. Ils entraînent un ResNet-50 [30] à prédire les notes données avec le système Veeck et Zahinovic [27], c’est-à-dire le développement des blastocystes, la qualité du TE et la qualité de la MCI.

4 Protocoles et modèles utilisés

Nous utilisons 953 vidéos d’embryons provenant de FIV réalisées entre 2011 et 2019 au laboratoire de biologie et de médecine de la reproduction du centre hospitalier universitaire de Nantes. Afin de réduire les temps de calculs, nous n’utilisons qu’un seul plan focal par embryon, qui est choisit automatiquement par l’Embryoscope pour chaque image de la vidéo. Nous utilisons les phases de développement évoquées en section 2, à savoir $tPB2$, $tPNa$, $tPNf$, $t2$, $t3$, $t4$, $t5$, $t6$, $t7$, $t8$, $t9+$, tM , tSB , tB et enfin tEB . Les modèles testés doivent donc résoudre un problème de classification à 15 classes.

Ce travail consiste à classer des images extraites de vidéos pour extraire la phase de développement correspondant à chaque image. Nous travaillons donc sur la même tâche que les travaux cités dans la section 3.1. Nous travaillons avec un plus grand nombre de phases que [13, 12, 11, 16, 17, 18] qui utilisent 6 phases ou moins. Nous profitons du fait que nous disposons déjà d’un grand nombre de vidéos où les 15 phases sont annotées et qu’il est possible que chacune de ces 15 phases soit importante pour prédire la qualité intrinsèque de l’embryon. Nous cherchons aussi à exploiter le contexte de chaque image, à savoir les images suivantes et précédentes, comme [16, 17, 18, 11]. Afin de simplifier le problème, nous ne cherchons pas à prédire la localisation de l’embryon ou des cellules, contrairement à [12, 18]. L’algorithme de Viterbi est utilisé pour rendre les prédictions du modèle cohérente tout au long de la vidéo, comme proposé dans [11, 18, 17, 16].

4.1 Modèles comparés

Plusieurs modèles ont été utilisés pour réaliser cette tâche de classification. Ils sont illustrés par la Figure 2 et détaillés ci-après.

Le modèle ResNet-18 [30] et plus généralement les modèles résiduels sont largement utilisés pour la classification d’image par exemple sur ImageNet [30]. Ce modèle est composé exclusivement de couches de convolutions (17 couches pour la version ResNet-18) et contient des connexions résiduelles toute les 2 couches. La résolution et le nombre de canaux des cartes de traits sont respectivement divisée et multiplié par deux toute les 4 couches. Après les convolutions, une couche de “average-pooling” permet d’obtenir un vecteur de traits, auquel on applique la couche “soft-max” finale pour faire la prédiction.

Le modèle ResNet-LSTM est la combinaison du modèle ResNet-18 avec un LSTM [19]. Le modèle LSTM a été conçu pour modéliser des séquences et a été appliqué avec succès dans des tâches telle que la reconnaissance

de la parole [31]. Les pré-activations de l’avant-dernière couche sont utilisées comme vecteur de traits et sont transmises à un LSTM bidirectionnel à deux couches qui modélise l’évolution à travers les étapes du temps. La taille de chaque unité cachée est 1024. Une couche linéaire au-dessus du LSTM calcule les scores.

Le modèle ResNet3D [32] est une variante de ResNet conçue pour la classification de séquences d’images. Ce modèle permet de modéliser la séquence d’images en fusionnant les informations temporelles au fur et à mesure qu’on avance en profondeur dans le réseau. Ce modèle permet donc une fusion à la fois tardive et précoce de l’information. Pour cette application, les paramètres de max-pooling et de stride sont fixés à 1 dans la dimension temporelle. La suppression de l’agrégation temporelle est nécessaire car nous voulons une prédiction pour chaque image. Nous utilisons la variante *r2plus1d_18* proposée dans [32].

5 Expériences

L’expérience principale de ce papier consiste à entraîner plusieurs architectures et à les évaluer à l’aide d’une validation croisée ($k = 10$). Les détails de l’expérience sont indiqués ci-dessous.

Méta-paramètres. Chaque batch est composé de 10 séquences de 4 images consécutives. La position de la séquence dans la vidéo est choisie aléatoirement dans la vidéo. La fonction de perte est optimisée avec la méthode de descente de gradient stochastique, avec un taux d’apprentissage constant de 0,001 et un momentum de 0,9. Nous appliquons du dropout [33] ($p = 0.50$) sur la dernière couche de chaque modèle pendant l’entraînement. Le modèle ResNet-18 traite chaque image de manière indépendante et lit donc les $10 \times 4 = 40$ images en parallèle, afin d’avoir un nombre d’images par batch égal pour les trois modèles et ainsi permettre une meilleure comparaison.

Lors de la validation, pour réduire l’occupation de la mémoire GPU, les modèles ne sont pas évalués sur l’ensemble de la vidéo en une seule fois, mais sur des séquences de 150 images. Comme chaque vidéo contient environ 500 images, quelques inférences suffisent pour analyser une vidéo entière.

Soit N le nombre total d’images d’entraînement et L le nombre d’images dans une séquence. Une époque se termine lorsque le modèle a vu N/L séquences. Pour sélectionner les séquences, nous utilisons des tirages aléatoires avec remise, c’est-à-dire que le modèle peut voir plusieurs fois la même image et peut ne pas voir certaines images au sein d’une époque.

Nous utilisons 88% des vidéos pour l’entraînement, 6% pour la validation et 6% pour le test. Le modèle est entraîné jusqu’à ce que les performances sur l’ensemble de validation ne s’améliorent pas pendant 30 époques consécutives, avec un maximum de 80 époques. Le meilleur modèle de l’ensemble de validation est alors restauré et évalué sur l’ensemble de test.

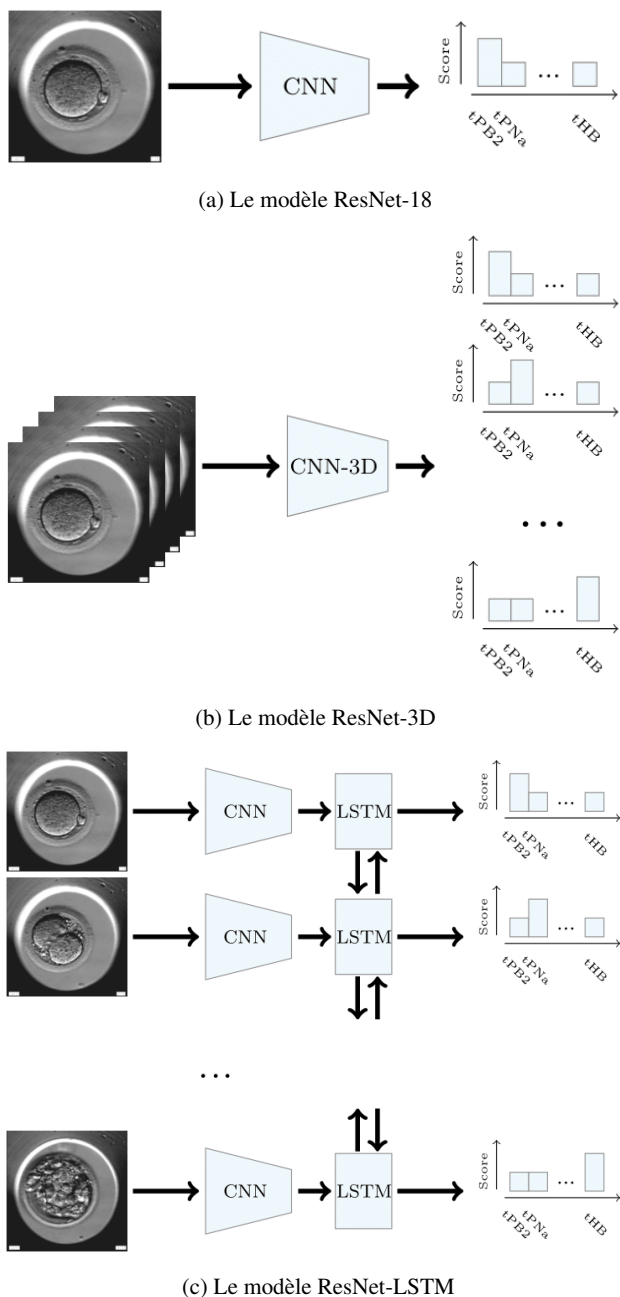


FIGURE 2 – Les différents modèles utilisés.

Pré-traitement et sélections des vidéos. Aucune augmentation de données n'est utilisée. Les images sont réduites de 500×500 à 224×224 pour réduire l'occupation de mémoire GPU.

Certains embryons se développent mal et leur croissance peut être interrompue par un biologiste. Lorsque cela se produit, étant donné que ces embryons ne seront pas transférés les biologistes ne les annotent pas aussi soigneusement que quand il s'agit d'embryons avec un développement normal. Afin de réduire le bruit dans les annotations, nous choisissons de ne garder que les vidéos montrant au moins 6 phases de développement distinctes. Le nombre de

vidéos final est de 788.

Initialisation des poids. Les poids du ResNet-18 et du ResNet3D sont respectivement pré-entraînés sur ImageNet [34] et Kinetics [35]. Les poids du LSTM sont initialisés de manière aléatoire.

Matériels et logiciels utilisés Les modèles ont été implémentés avec la bibliothèque Pytorch 1.4.0 [36]. Une partie des modèles a été entraînés sur deux cartes Tesla P100 (32 Go de mémoire graphique au total) et, pour des raisons de temps, le reste des modèles a été entraîné sur CPU.

5.1 Métriques

Plusieurs métriques sont utilisées pour évaluer les modèles. La précision p est une des métriques les plus utilisées en classification d'images et se calcule par la proportion d'images correctement étiquetées par le modèle. Nous calculons également cette métrique en utilisant l'algorithme de Viterbi au préalable afin que les prédictions soient cohérentes tout au long de la vidéo, comme il est souvent fait dans la littérature [11, 18, 17, 16]. Cette deuxième métrique est notée p_v . La précision temporelle p_t est la proportion moyenne des transitions de phase qui sont prédites suffisamment proches de la transition réelle correspondante. Par "assez proche", nous entendons que le temps de la transition prédite ne se situe pas à plus d'une heure du temps de la transition réelle. Elle exige donc que les prédictions soient rendues cohérentes en utilisant Viterbi. Une métrique similaire est utilisé dans [37]. Enfin, la corrélation linéaire c entre le temps de transition prédit et le temps réel de la transition correspondante. Elle nécessite donc également Viterbi.

5.2 Résultats

Dans le Tableau 1 sont indiquées les performances moyennes avec écart type après 10 validations croisées et dans le Tableau 2 sont indiqués les p-valeurs montrant la signifiacance statistique des écarts entre les performances moyennes. Les valeurs moyennes de la corrélation linéaire sont très proche de 1 probablement parce que l'algorithme de Viterbi impose d'avoir des prédictions cohérentes et donc une transition entre deux phases tardives sera toujours prédite après une transition entre deux phases au début du développement. Cela implique que les instants auxquels les modèles prédisent une transition sera toujours une fonction monotone croissante du temps, ce qui biaise la corrélation vers 1.

L'architecture ResNet-3D permet un gain significatif de performance par rapport à ResNet-18 et ResNet-LSTM. Le fait que le modèle ResNet-LSTM ait une performance qui ne soit pas significativement supérieure à celle de ResNet indique qu'il faudrait peut-être augmenter la capacité de la partie LSTM de ce modèle (augmentation du nombre de couches, de la taille des unités cachées) ou la longueur des séquences traitées. Il est intéressant de noter que même sur des séquences très courtes (4 images), ResNet-3D arrive à obtenir des performances significativement meilleures que ResNet-18. Cela indique que le gain de performance pourrait éventuellement être encore meilleur sur des séquences

| Modèle | p | p_v | c | p_t |
|-------------|-----------------------------------|-----------------------------------|---------------------------------------|-----------------------------------|
| ResNet | 0.7 ± 0.03 | 0.72 ± 0.03 | 0.9917 ± 0.0018 | 0.51 ± 0.04 |
| ResNet-3D | 0.74 ± 0.03 | 0.77 ± 0.03 | 0.9934 ± 0.0011 | 0.61 ± 0.02 |
| ResNet-LSTM | 0.71 ± 0.04 | 0.72 ± 0.04 | 0.9913 ± 0.0023 | 0.52 ± 0.05 |

TABLE 1 – Les performances des modèles évalués. Les valeurs indiquées sont les moyennes accompagnées des écart-types.

| Métrique | Modèle | ResNet | ResNet-3D |
|----------|-------------|----------------|----------------|
| p | ResNet-3D | 0.01 | |
| | ResNet-LSTM | 0.43 | 0.1 |
| p_v | ResNet-3D | 0.01 | |
| | ResNet-LSTM | 0.8 | 0.02 |
| c | ResNet-3D | 0.02 | |
| | ResNet-LSTM | 0.67 | 0.02 |
| p_t | ResNet-3D | 5e – 06 | |
| | ResNet-LSTM | 0.63 | 0.00013 |

TABLE 2 – p-valeurs obtenues suite à la validation croisée. Le modèle ResNet-3D est significativement meilleur que ResNet-18 selon toutes les métriques et que ResNet-LSTM selon trois métriques. Le gain apporté par ResNet-LSTM par rapport à ResNet-18 n'est pas significatif.

plus longues. Les métriques p et p_v ont le défaut de pénaliser autant les modèles qui proposent des transitions de phases loin de vraies transitions que ceux qui se trompent de quelques images seulement. La métrique de précision temporelle prend en compte cet aspect : un modèle qui prédit un changement de phase près du changement de phase réel est favorisé par rapport à un modèle qui est loin de la vérité. Cette métrique montre que la performance du ResNet-3D est largement supérieure à celle de ResNet ou ResNet-LSTM (61% contre 51% ou 52%). Le gain de performance de ResNet-3D est donc sous-estimé si l'on se réfère seulement aux métriques de précision ou précision temporelle.

La figure 3 illustre bien en quoi ResNet-3D est meilleur que ResNet-18. Ce dernier produit des prédictions qui sont instables alors que ResNet-3D parvient à faire des prédictions plus cohérentes, probablement parce qu'il agrège les images au niveau temporel.

6 Travaux futurs

Les expériences faites dans ce travail n'utilise qu'un seul plan focal par image, ce qui empêche d'avoir une bonne vision de certaines parties du puits dans lequel se trouve l'embryon (i.e les zones situées au dessus et en dessous du plan focal). Les Embryoscopes capturent chaque image en utilisant tous les plans focaux ce qui produit des images 3D. Utiliser les images provenant de tous les plans focaux permettrait probablement d'améliorer la performance. Il serait aussi intéressant de forcer le modèle à se concentrer sur une partie de l'image afin de rendre sa décision plus interprétable comme par exemple avec de l'attention dure dans [18] ou avec de l'attention douce dans [38]. L'attention douce ne réduit pas la taille de l'image traitée mais permet de mettre en valeur n'importe quel détail dans l'image,

peu importe sa forme, ce qui améliore l'interprétabilité. Dans le futur nous souhaitons exploiter de plus longues séquences d'images et donc mieux tirer partie des systèmes avec LSTM, mais il faut résoudre le problème d'occupation mémoire GPU que cela pourrait entraîner. Enfin, nous allons utiliser les paramètres MC prédits par notre modèle ainsi que d'autres variables biologiques pour prédire le taux de grossesse.

Références

- [1] Marcia C. Inhorn and Pasquale Patrizio. Infertility around the globe : new thinking on gender, reproductive technologies and global movements in the 21st century. *Human Reproduction Update*, 21(4) :411–426, 03 2015.
- [2] A. P. Ferraretti, V. Goossens, M. Kupka, S. Bhattacharya, J. de Mouzon, J. A. Castilla, K. Erb, V. Korsak, A. Nyboe Andersen, and European IVF-Monitoring (EIM) Consortium for the European Society of Human Reproduction and Embryology (ESHRE). Assisted reproductive technology in Europe, 2009 : results generated from European registers by ESHRE. *Human Reproduction (Oxford, England)*, 28(9) :2318–2331, September 2013.
- [3] S. Dyer, G.M. Chambers, J. de Mouzon, K.G. Nygren, F. Zegers-Hochschild, R. Mansour, O. Ishihara, M. Banker, and G.D. Adamson. International Committee for Monitoring Assisted Reproductive Technologies world report : Assisted Reproductive Technology 2008, 2009 and 2010†. *Human Reproduction*, 31(7) :1588–1609, 05 2016.
- [4] Joe Conaghan, Alice A. Chen, Susan P. Willman, Kristen Ivani, Philip E. Chenette, Robert Boostanfar, Valerie L. Baker, G. David Adamson, Mary E. Abusief, Marina Gvakharia, Kevin E. Loewke, and Shehua Shen. Improving embryo selection using a computer-automated time-lapse image analysis test plus day 3 morphology : results from a prospective multicenter trial. *Fertility and Sterility*, 100(2) :412–419.e5, Aug 2013.
- [5] pubmeddev and Pribenszky C. al, et. Time-lapse culture with morphokinetic embryo selection improves pregnancy and live birth chances and reduces early pregnancy loss : a meta-analysis. - PubMed - NCBI.
- [6] Minghao Chen, Shiyu Wei, Junyan Hu, Jing Yuan, and Fenghua Liu. Does time-lapse imaging have fa-

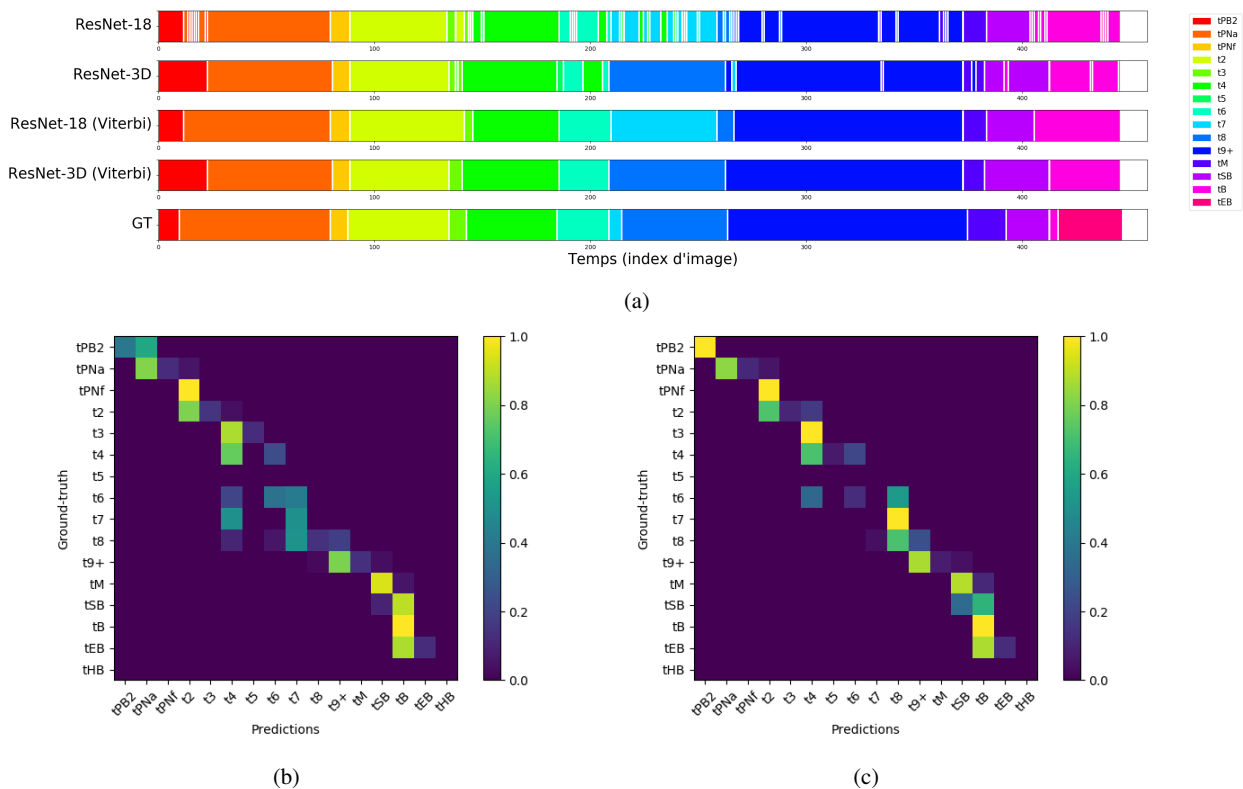


FIGURE 3 – Visualisation des prédictions faites par un modèle ResNet-18 et un modèle ResNet-3D sur une vidéo. Les modèles ont eu le même jeu d'entraînement, de validation et de test. La vidéo analysée ici fait partie du jeu de test. 3a) Une frise montrant les prédictions faites les deux modèles. Les deux lignes du haut montrent les prédictions brutes du modèles, les deux lignes suivantes montrent les prédictions obtenues après avoir appliqué l'algorithme Viterbi sur les prédictions brutes. La dernière ligne montre la vérité terrain et chaque couleur représente une phase de développement. Sans utiliser Viterbi, le modèle ResNet-3D donne des prédictions assez stables d'une image à l'autre alors que ResNet-18 change régulièrement de prédiction. La figure est mieux visible en couleur. En 3b) et 3c) sont montrées respectivement les matrices de confusion de ResNet-18 et ResNet-3D

- vorable results for embryo incubation and selection compared with conventional methods in clinical in vitro fertilization? A meta-analysis and systematic review of randomized controlled trials. *PLoS One*, 12(6) :e0178720, 2017.
- [7] Bjørn Molt Petersen, Mikkel Boel, Markus Montag, and David K. Gardner. Development of a generally applicable morphokinetic algorithm capable of predicting the implantation potential of embryos transferred on Day 3. *Human Reproduction (Oxford, England)*, 31(10) :2231–2244, 2016.
- [8] Robert Milewski, Agnieszka Kuczyńska, Bożena Stankiewicz, and Waldemar Kuczyński. How much information about embryo implantation potential is included in morphokinetic data? a prediction model based on artificial neural networks and principal component analysis. *Advances in Medical Sciences*, 62(1) :202 – 206, 2017.
- [9] Linda Sundvall, Hans Ingerslev, Ulla Knudsen, and Kirstine Kirkegaard. Inter- and intra-observer variability of time-lapse annotations. *Human reproduction (Oxford, England)*, 28, 09 2013.
- [10] H. Nadir Ciray, Alison Campbell, Inge Errebo Agerholm, Jesús Aguilar, Sandrine Chamayou, Marga Esbert, and Shabana Sayed. Proposed guidelines on the nomenclature and annotation of dynamic human embryo monitoring by a time-lapse user group. *Human Reproduction*, 29(12) :2650–2660, December 2014.
- [11] Aisha Khan, Stephen Gould, and Mathieu Salzmann. Deep convolutional neural networks for human embryonic cell counting. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 339–348, Cham, 2016. Springer International Publishing.
- [12] R. M. Rad, P. Saeedi, J. Au, and J. Havelock. Blastomere cell counting and centroid localization in microscopic images of human embryo. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6, Aug 2018.
- [13] Manoj Kumar Kanakasabapathy, Prudhvi Thirumalaraju, Charles L. Bormann, Hemanth Kandula, Irene Dimitriadis, Irene Souter, Vinish Yogesh, Sandeep Kota Sai Pavan, Divyank Yarravarapu, Raghav Gupta, Rohan Pooniwal, and Hadi Shafiee. Development and evaluation of inexpensive automated deep learning-based imaging systems for embryology. *Lab Chip*, 19 :4139–4145, 2019.
- [14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.
- [15] François Chollet. Xception : Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016.
- [16] Nathan H Ng, Julian McAuley, Julian A Gingold, Nina Desai, and Zachary C Lipton. Predicting embryo morphokinetics in videos with late fusion nets & dynamic decoders, 2018.
- [17] Z. Liu, B. Huang, Y. Cui, Y. Xu, B. Zhang, L. Zhu, Y. Wang, L. Jin, and D. Wu. Multi-task deep learning with dynamic programming for embryo early development stage classification from time-lapse videos. *IEEE Access*, 7 :122153–122163, 2019.
- [18] Tingfung Lau, Nathan Ng, Julian Gingold, Nina Desai, Julian J. McAuley, and Zachary C. Lipton. Embryo staging with weakly-supervised region selection and dynamically-decoded predictions. *CoRR*, abs/1904.04419, 2019.
- [19] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8) :1735–1780, November 1997.
- [20] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN : towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.
- [21] A. Singh, J. Au, P. Saeedi, and J. Havelock. Automatic segmentation of trophectoderm in microscopic images of human blastocysts. *IEEE Transactions on Biomedical Engineering*, 62(1) :382–393, Jan 2015.
- [22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.
- [23] S. Kheradmand, A. Singh, P. Saeedi, J. Au, and J. Havelock. Inner cell mass segmentation in human hmc embryo images using fully convolutional network. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1752–1756, Sep. 2017.
- [24] R. M. Rad, P. Saeedi, J. Au, and J. Havelock. Multi-resolutional ensemble of stacked dilated u-net for inner cell mass segmentation in human embryonic images. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 3518–3522, Oct 2018.
- [25] José Celso Rocha, Felipe José Passalia, Felipe Delestro Matos, Maria Beatriz Takahashi, Marc Peter Maserati Jr, Mayra Fernanda Alves, Tamie Guibu de Almeida, Bruna Lopes Cardoso, Andrea Cristina Basso, and Marcelo Fábio Gouveia Nogueira. Automated image processing of bovine blastocysts produced in vitro for quantitative variable determination. *Scientific Data*, 4 :170192 EP –, Dec 2017. Data Descriptor.
- [26] Pegah Khosravi, Ehsan Kazemi, Qiansheng Zhan, Jonas E. Malmsten, Marco Toschi, Pantelis Zisimopoulos, Alexandros Sigaras, Stuart Lavery, Lee A. D. Cooper, Cristina Hickman, Marcos Meseguer, Zev Rosenwaks, Olivier Elemento, Nikica Zaninovic, and

- Iman Hajirasouliha. Deep learning enables robust assessment and selection of human blastocysts after in vitro fertilization. *npj Digital Medicine*, 2(1) :21, 2019.
- [27] Lucinda L. Veeck and Nikica Zaninovic. An Atlas of Human Blastocysts. Library Catalog : www.crcpress.com.
- [28] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015.
- [29] Tsung-Jui Chen, Wei-Lin Zheng, Chun-Hsin Liu, Ian Huang, Hsing-Hua Lai, and Mark Liu. Using deep learning with large dataset of microscope images to develop an automated embryo grading system. *Fertility & Reproduction*, 01(01) :51–56, 2019.
- [30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.
- [31] A. Graves, N. Jaitly, and A. Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 273–278, Dec 2013.
- [32] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3d residual networks for action recognition. *CoRR*, abs/1708.07632, 2017.
- [33] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15 :1929–1958, 2014.
- [34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3) :211–252, 2015.
- [35] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.
- [36] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.
- [37] M Feyeux, Arnaud Reignier, M Mocaer, J Lammers, Dimitri Meistermann, P Barrière, Perrine Paul, Laurent David, and Thomas Fréour. Development of automated annotation software for human embryo morphokinetics. *Human reproduction (Oxford, England)*, 03 2020.
- [38] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better : Recurrent attention convolutional neural network for fine-grained image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Apprentissage profond et localisation d'objets : deux exemples d'applications en agriculture

L. Rigal^{1,2}, J. Chopin¹, J.-B. Fasquel¹, R. Fernandez², C. Challe-Blanloeil²

¹ LARIS, EA7315, Université d'Angers, Angers, France

² Terrena, Digital Lab, La Noëlle, 44150 Ancenis, France

louis.rigal@etud.univ-angers.fr

Résumé

Cet article concerne la vision par ordinateur en agriculture, en se focalisant sur la localisation d'objets par apprentissage profond. Dans ce cadre, nous présentons d'abord les réseaux de neurones *Faster R-CNN* et *YOLO*, en détaillant ce dernier, reconnu pour sa rapidité de traitement. Nous illustrons ensuite leur potentiel pour deux applications originales : la qualification de la pododermatite des poulets en abattoir, et le suivi et comptage de porcs. Les résultats obtenus montrent l'intérêt de l'apprentissage profond dans un monde agricole en pleine mutation.

Mots-clés

Vision par ordinateur, Apprentissage Profond, Agriculture.

Abstract

This paper concerns computer vision in agriculture, by focusing on object localization by means of deep learning. We first present two recent deep neural networks dedicated to object detection, namely *Faster R-CNN* and *YOLO*, and provide a detailed description of the latter one. We then illustrate their potential for two original applications : the qualification of pododermatitis on chicken feet in slaughterhouses, as well as the tracking and counting of pigs from video streams. Our results demonstrate the relevance of deep learning in a changing agricultural world.

Keywords

Computer vision, Deep learning, Agriculture.

1 Introduction

L'intelligence artificielle et plus particulièrement l'apprentissage automatique offre de multiples perspectives dans un domaine agricole en pleine mutation [9].

Dans ce contexte, nous développons des applications répondant à des besoins du monde agricole, notamment autour de la vision, en s'appuyant sur ces outils. Dans cet article, nous nous intéressons à deux applications originales requérant la localisation et la reconnaissance d'objets : la détection de la pododermatite [15] dans les abattoirs de poulets et le comptage de cochons [10]. Ces deux applications reposent sur les réseaux profonds "Faster R-CNN" et "Yolo" [5, 12], récemment introduits pour cette probléma-

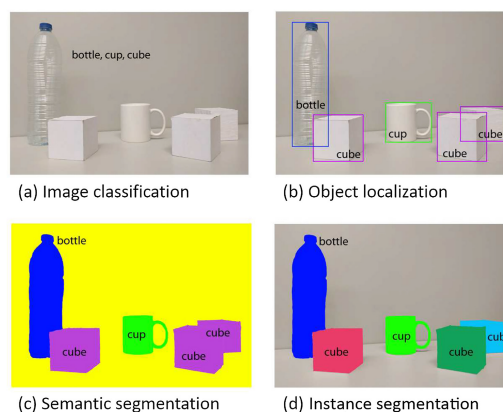


FIGURE 1 – Apprentissage profond et cas d'usages en vision [4] : classification (a), détection (b), segmentation sémantique (c) et segmentation d'instances (d).

tique de localisation et reconnaissance en vision par ordinateur. Ces types de réseaux ont été proposés dans la continuité des travaux en apprentissage profond [6], initialement focalisés sur la tâche de classification. Sous réserve d'une base d'entraînement conséquente, l'apprentissage profond permet notamment de s'affranchir des étapes couramment considérées auparavant de segmentation, caractérisation et classification d'objets, impliquant des algorithmes spécifiques souvent complexes à concevoir.

La première partie de cet article, associée à la section 2, présente la problématique de la reconnaissance et de la localisation d'objet à base de réseaux de neurones profonds, en détaillant le principe de la plus récente des deux architectures, à savoir Yolo. La seconde partie (section 3) traite des deux applications considérées, reposant sur cette classe commune de réseaux.

2 Localisation d'objets par apprentissage profond

L'apprentissage automatique et en particulier l'apprentissage profond ont montré leur efficacité dans de nombreux domaines [6]. La figure 1 en illustre des usages courants en vision par ordinateur [4]. Il existe quatre tâches de vision

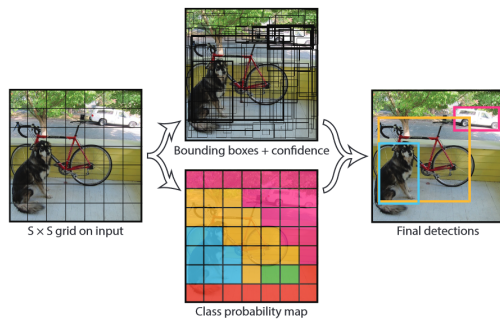


FIGURE 2 – Schéma du modèle de Yolo, pour $S = 7$, issu de [12]

principales que les algorithmes de vision cherchent à réaliser. La classification, la plus classique, est celle de savoir si un objet particulier est présent dans une image (on affecte une classe à l'image entière). La détection (impliquant la localisation), consiste à localiser et classifier les différents objets présents dans une image (souvent en les inscrivant dans une boîte englobante (*Bounding Box*)). Enfin, la segmentation implique de classifier chaque pixel de l'image comme appartenant à un type d'objet particulier. On fait alors la différence entre la segmentation sémantique, qui attribue la même classe à deux objets du même type, et la segmentation par instance, qui distingue différents objets (instances) d'une même classe. Nous nous intéressons ici à la tâche de détection et de localisation par apprentissage profond : dans ce contexte, de récents travaux ont permis de proposer les architectures Faster R-CNN et YOLO ("You Only Look Once"), à base de réseaux de neurones profonds, efficaces pour l'identification et la localisation simultanée d'objets [5, 12]. Pour une détection et localisation plus rapide, l'architecture YOLO [12] (et sa récente version V3 [14]) semble plus adaptée que le Faster R-CNN, même si ces deux algorithmes présentent certaines similarités [12].

Nous donnons ci-après quelques éléments autour du réseau YOLO [12, 14], comprenant l'étape de détection de boîtes englobantes candidates (section 2.1), puis l'étape de filtrage de celles-ci afin d'obtenir la plus pertinente (section 2.2).

2.1 Détection des boîtes englobantes par le réseau YOLO

Pour détecter les objets d'une image, un réseau YOLO commence par diviser celle-ci en un damier de cellules. Prenons par exemple un damier de $S \times S$ cellules (la figure 2 donne un exemple dans le cas où $S = 7$). Dans chacune de ces cellules, le réseau prévoit plusieurs boîtes englobantes (Bounding Boxes ou BB), destinées à encadrer de potentiels objets. Une seule évaluation de l'image permet à la fois de prédire la position et la taille optimale de chaque BB pour chaque cellule, ainsi que la probabilité de chacune d'elles de contenir un objet (figure 2).

En sortie, le réseau produit un tenseur $T \in \mathbb{R}^{S \times S \times D}$, où

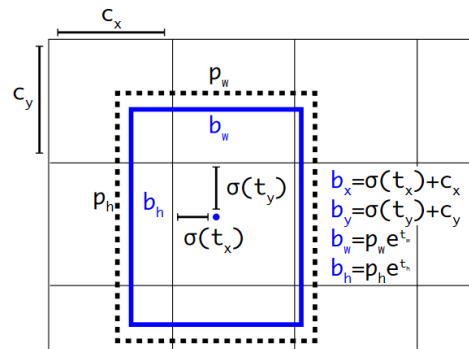


FIGURE 3 – Définition de la Bounding Box, issue de [13]



FIGURE 4 – Principe de Non-Maximum Suppression

$D = B \times 5 + C$. Ici, B désigne le nombre de BB calculées pour chaque cellule, et C le nombre de classes d'objets que le réseau doit être apte à reconnaître. A noter que le coefficient 5 (terme $B \times 5$) signifie que chaque BB est caractérisée par 5 valeurs décrites ci-après.

Comme le montre la figure 3, pour chaque BB le réseau donne des positions b_x et b_y relatives au coin supérieur gauche de la cellule à laquelle elle appartient, une largeur b_w et une hauteur b_h , ainsi qu'un score de *confiance* défini lors de l'entraînement par :

$$Conf = Pr(Object) * IoU_{pred}^{truth} \quad (1)$$

soit la probabilité que la BB contienne un objet ($Pr(Object)$), multipliée par sa correspondance avec la vérité terrain en terme de surface recouverte. On rappelle la définition mathématique de l'*IoU* ("Intersection Over Union") :

$$IoU_{pred}^{truth} = \frac{BB_{truth} \cap BB_{pred}}{BB_{truth} \cup BB_{pred}} \quad (2)$$

Enfin, pour chaque cellule, le réseau donne C scores d'appartenance de l'objet contenu dans la cellule à une classe (terme C de la relation précédente $B \times 5 + C$), avec :

$$appart_j^i = Pr(Class_i | Object_j) \quad (3)$$

où $appart_j^i$ représente la probabilité que l'objet de la cellule j appartienne à la classe i . Ces scores d'appartenance sont donc partagés par toutes les BB originaires d'une même cellule.

Dans les dernières versions de YoLo [13, 14], les BB ont des formats particuliers par défaut. Ces formats sont appelés *Anchor Boxes* ou Boîtes d'ancrage (BA). Elles sont

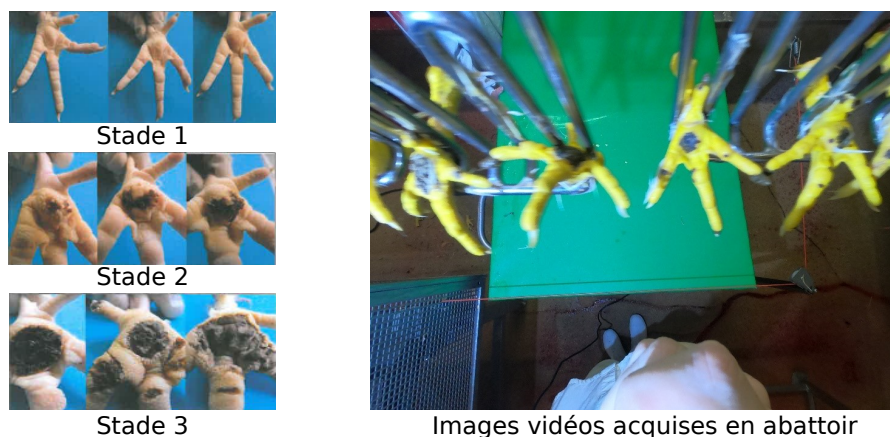


FIGURE 5 – Pododermatite : les trois stades de la maladie (gauche) et l’environnement de détection en abattoir (droite). Le stade 1 sera confondu avec les pattes saines, l’enjeu principal étant de détecter les stades 2 (lésions superficielles) et 3 (lésions avec ulcération).

issues du constat qu’au sein d’un même jeu de données, on peut déterminer un nombre réduit de formats de BB qui correspondraient correctement à un bon nombre d’objets. Par exemple, si le détecteur d’objets n’est utilisé que sur des piétons, une BB avec un format plutôt vertical a des chances de correspondre à de nombreux exemples, tandis qu’une BB horizontale ne va jamais servir. On choisit donc de déterminer un jeu de quelques formats de BB, qui servent de base lors de la détection de l’objet : on part de ces formats standards, et pas d’un ratio hauteur-largeur aléatoire, pour trouver le format réel de l’objet.

En pratique, on réalise une étude sur notre jeu de données, et en utilisant un algorithme de *clustering* (e.g., *K-Means*) sur l’ensemble des différents formats, on trouve les BA qui représentent le mieux notre problème.

Cette détermination de BA s’accompagne d’une multiplication des formats de cellules découpant l’image. La multiplication des *damiens* a pour but de réaliser des détections à plusieurs échelles différentes : les cellules les plus larges sont chargées de détecter les objets les plus larges, et se voient donc attribuer par défaut les BA les plus étendues, et inversement pour les objets de petite taille.

2.2 Suppression des Non-Maximas

Les modèles de détection d’objets génèrent donc une multitude de propositions de BB, toutes évaluées par un classifieur. On peut imaginer que beaucoup de ces BB se référeront au même objet dans l’image, comme peut le montrer la Figure 4, extraite du site “towardsdatascience”¹. Il faut donc intégrer à la chaîne de traitement un mécanisme qui exclut la plupart des BB pour n’en garder qu’une par objet, la meilleure.

Pour ce faire, on recourt à un algorithme de Suppression des Non-Maximas (NMS). L’algorithme consiste en cette suite d’étapes :

En entrée : Une liste **B** de BB proposées pour une même

classe d’objet, associées à leur score d’appartenance à cette classe *appart*, introduit précédemment (3), et un seuil global de chevauchement **N**. Une même BB n’entre que dans l’algorithme de NMS qui concerne la classe d’objet pour laquelle son score *appart* est le plus élevé.

En sortie : Une liste de BB filtrée **D**

1. On sélectionne la BB avec le plus haut score *appart*, on l’enlève de **B** et on la rajoute à **D**
2. On compare ensuite cette BB avec toutes les autres contenues dans **B**, celles qui ont un *IoU* supérieur à **N** sont retirées de **B**
3. Répéter ces étapes avec la prochaine BB avec le plus haut score d’appartenance. On continue jusqu’à avoir vidé **B**

On répète cet algorithme pour toutes les classes d’objets. Différents choix de **N** peuvent mener à différents résultats. Un **N** trop élevé va faire que le modèle sélectionne plusieurs BB pour un même objet. Un **N** trop faible empêchera le système de générer des BB pour les objets trop proches. Le seuil **N** est déterminé arbitrairement pendant et après l’entraînement. On peut typiquement choisir un seuil d’*IoU* de 0.6 pour **N**.

3 Applications

Nous présentons ci-après deux applications des réseaux présentés précédemment : la détection de la pododermatite chez les poulets et le comptage de cochons. Ces applications sont abordées ci-après, en précisant le contexte, l’approche considérée (intégrant les Faster R-CNN ou Yolo), ainsi que les expériences menées (préliminaires dans le cas de la seconde application).

3.1 Détection de la pododermatite chez le poulet

Cette application concerne la problématique du bien-être animal, et plus précisément de la détection de la pododermatite [15] chez le poulet. Il s’agit d’une maladie que

1. [https://towardsdatascience.com/\\$\at\\$k.sambasivarao222](https://towardsdatascience.com/\atk.sambasivarao222), consulté en Mars 2020.

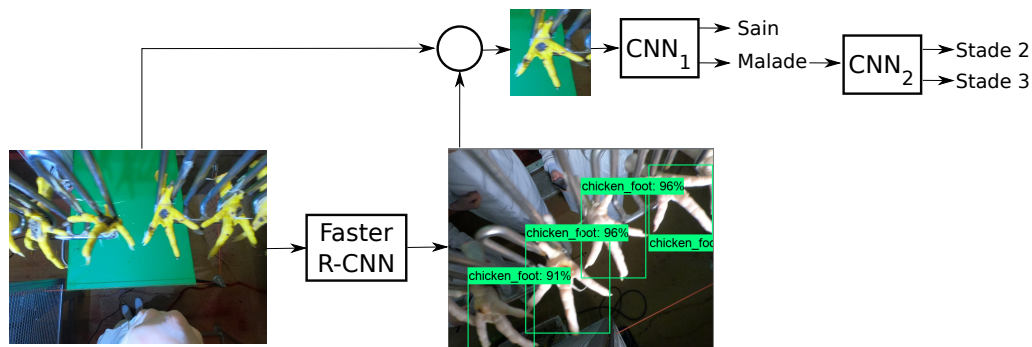


FIGURE 6 – Architecture de traitement considérée pour détecter les pattes puis les classer.

se traduit par l'apparition de lésions au niveau des pattes, que l'on peut classer en 3 stades selon la grille de notation ITAVI (voir la figure 5-gauche pour les différents stades). L'objectif de ce travail est le développement d'un système de vision, fonctionnant en abattoir, permettant la détection des pattes malades et, le cas échéant, du stade de la maladie (voir la figure 5-droite pour l'environnement de traitement). Une contrainte importante est la vitesse de traitement, qui doit être compatible avec la cadence élevée de défilement des pattes en abattoir (8000 à 11000 poulets/heure). A notre connaissance, la détection de la pododermatite chez le poulet par vision par ordinateur n'a jamais été abordée dans la littérature scientifique.

Dans notre contexte, le stade 1 (absence de lésions, ou lésions minimales) sera assimilé au cas sain. L'enjeu est donc principalement de détecter les pattes, puis de les classer dans l'une des trois catégories : sain (ou stade 1), stade 2 et stade 3.

La figure 6 donne un schéma de l'architecture qui a été implémentée pour traiter ce problème, en exploitant les récents travaux de recherche portant sur la localisation d'objets par apprentissage profond, présentés précédemment. La détection des pattes est gérée par un réseau du type "Faster R-CNN". Une fois les pattes détectées, elles sont classées en utilisant deux réseaux de type CNN ("Convolutional Neural Network") pour séparer les pattes saines des pattes malades (CNN1 sur la figure 6) puis distinguer le stade de pododermatite (CNN2 sur la figure 6). Pour ces deux étapes de classification, le réseau VGG 16, pré-entraîné sur la base ImageNet, a été utilisé [4]. A noter que les résultats obtenus avec un seul réseau convolutif multi-classes (i.e. pour classer dans les trois catégories) se sont avérés moins bons. Il est probable que ceci provienne du déséquilibre entre les quantités de données représentant les 3 classes (tableau 1) : les stades 2 et 3 sont moins largement représentés.

Un point important a concerné l'entraînement des différents réseaux, nécessitant l'acquisition et l'annotation d'un grand nombre d'images, même si nous avons par ailleurs utilisé les techniques standard d'apprentissage par transfert ("transfert learning") et d'augmentation de données ("data augmentation") [4].

Pour la détection par Faster R-CNN, 500 images ont été

annotées manuellement en sélectionnant les boîtes englobantes des différentes pattes (chaque image contenant plusieurs pattes). Un ensemble de 13378 images de pattes (extraites avec la Faster R-CNN) a été utilisé pour la classification par CNN. Un grand nombre se sont avérées inclassifiables (e.g., mauvaise qualité) : le tableau 1 donne le nombre d'images finalement disponible par catégorie. L'annotation a été faite manuellement avec un logiciel conçu spécifiquement, par des experts pour les stades 2 et 3.

| | Sain/Stade 1 | Stade 2 | Stade 3 |
|-----------------|--------------|---------|---------|
| Nombre d'images | 5809 | 1636 | 2752 |

TABLE 1 – Données annotées manuellement et utilisables pour l'entraînement, la validation et le test.

Les tableaux 2 et 3 donnent respectivement les performances obtenues en sortie des CNN1 et CNN2, exprimées avec les métriques standard de précision (P - e.g., par rapport à la catégorie "saine" : capacité à classer une patte réellement saine dans la catégorie "saine") et rappel (R - e.g., par rapport à la catégorie "saine" : capacité à ne pas classer une patte malade dans la classe "saine" à tort). Le score F_1 est également indiqué, combinant les deux métriques précédentes :

$$F_1 = 2 \frac{P \times R}{P + R} \quad (4)$$

Les résultats montrent qu'il est difficile de correctement distinguer les stades 2 des stades 3 avec cette approche (voir les performances limitées en terme de précision, rappel et score F_1 rapportées par le tableau 3). Par contre, les performances s'avèrent très satisfaisantes pour distinguer les pattes saines des pattes malades, ceci étant l'objectif principal. A noter qu'en observant visuellement les résultats, il est apparu que les erreurs de classification entre les pattes saines et malades provenaient principalement des conditions d'acquisition (e.g., présence de paille masquant les lésions, terre "sombre" sous les ongles).

Suite à ces travaux, l'approche proposée est apparue tout à fait pertinente pour la détection des pattes malades (i.e., partie de l'architecture sans le CNN2), et le système a été

déployé dans plusieurs abattoirs gérés par Terrena. On accède ainsi à une estimation quantitative de cet indicateur du bien-être animal.

| | P | R | F_1 |
|--------|------|------|-------|
| Sain | 0.99 | 0.87 | 0.91 |
| Malade | 0.89 | 0.99 | 0.91 |

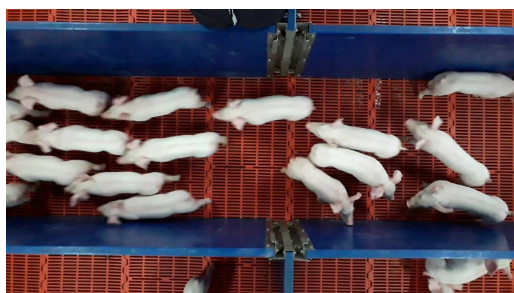
TABLE 2 – Performance de classification entre les pattes saines et malades (P , R et F_1 désignent respectivement la précision, le rappel et le score F_1).

| | P | R | F_1 |
|---------|------|------|-------|
| Stade 2 | 0.33 | 0.40 | 0.36 |
| Stade 3 | 0.76 | 0.36 | 0.49 |

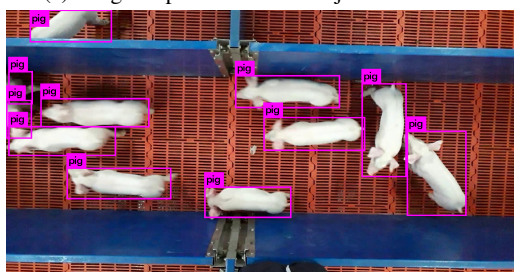
TABLE 3 – Performance de classification entre les différents stades (P , R et F_1 désignent respectivement la précision, le rappel et le score F_1).

3.2 Suivi et comptage de porcs

Cette application porte sur le comptage des porcs en transit dans un couloir (exemple en figure 7a). Si le nombre de cochons effectivement transités n’est pas rigoureusement suivi, cela peut être source de litiges entre les différents intervenants. C’est une situation qui arrive assez fréquemment, ainsi le développement d’une méthode de comptage automatique et fiable est souhaitable.



(a) Image de porcelets issue du jeu de données.



(b) Résultat de la détection par YOLO.

FIGURE 7 – Processus de détection des cochons.

Il existe des travaux connexes dans la communauté scientifique. Par exemple, il a récemment été fait mention de

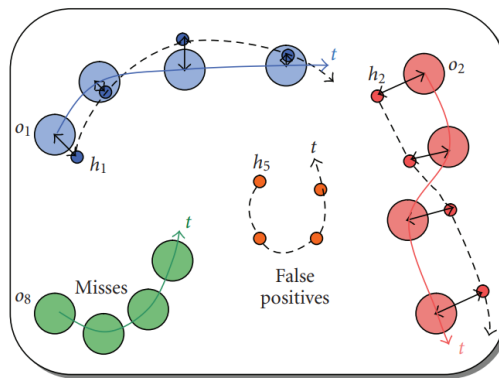


FIGURE 8 – Différents scénarii de suivi des objets, issu de [1]

la problématique de la reconnaissance de pores par vision pour s’affranchir de puces RFID [9]. Une autre contribution importante est celle de Psota et al. [11]. L’auteur met en place une architecture de réseau de neurones capable de faire une estimation de la localisation et de la position des cochons dans une image, par localisation et appariement des différentes parties de leurs corps. Cette architecture nous semble cependant trop imposante pour notre application, et l’estimation de la position des différentes parties du corps inutile. A noter qu’ils ont également publié un jeu de données conséquent de plus de 2000 images annotées de porcs dans différents environnements, que nous pourrions mettre à profit par la suite.

Un autre article récent sur le comptage de porcs a récemment été proposé, mais sans s’intéresser à la problématique du suivi. Les cochons sont comptés par intégration d’une carte de chaleur générée par un réseau de neurones [10]. Cette méthode a pour désavantage l’absence de localisation précise des cochons dans l’image, qui sont confondus en zones de forte densité. Dans ce cas d’utilisation, le nombre de cochons doit être estimé dans une zone cloisonnée à un instant précis, sans prendre en compte leurs mouvements ni leurs positions.

Dans notre cas, nous souhaitons pouvoir suivre les déplacements des cochons d’un bout à l’autre de la zone d’intérêt, en analysant les images d’un flux vidéo. Nous disposons de quelques vidéos filmées dans une même exploitation, dont la figure 7a est tirée. Le principe du suivi d’objets est de pouvoir détecter l’ensemble des objets d’un flux vidéo, et de pouvoir suivre individuellement leurs parcours sans les confondre ni en perdre la trace. Différents scénarii possibles lors du suivi d’objets sont représentés sur la figure 8. Il y manque le scénario d’interversion (ou de *mismatch*), qui survient lorsque deux objets trop proches échangent leurs identités entre des images consécutives du flux vidéo. Plusieurs travaux concernant le suivi d’objets sur un flux vidéo ont déjà été publiés [2, 7]. La plupart des systèmes de suivi décrits décomposent la tâche en ces quelques étapes :

1. **Détection** : Détection des objets, réalisée sur chaque image.

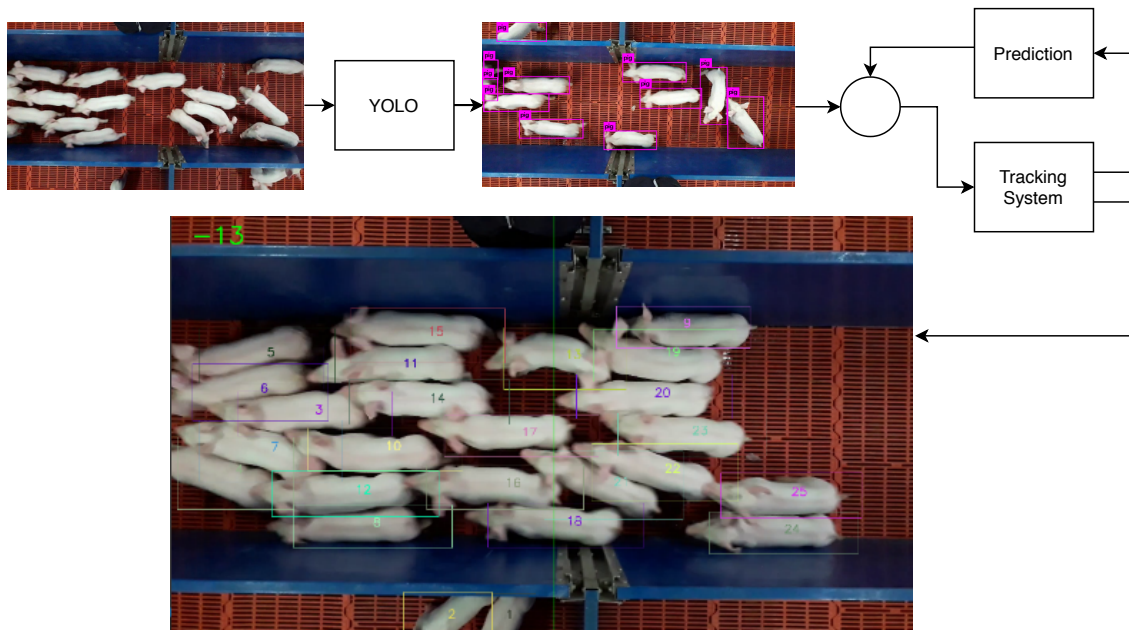


FIGURE 9 – Schéma complet du fonctionnement du système. Les détections sur les images brutes sont assurées par le réseau Yolo. Ces détections sont comparées aux prédictions réalisées par le système de tracking à l'image précédente. Le système de tracking associe les porcs suivis aux détections et attend la prochaine image.

2. **Extraction de caractéristiques** : On cherche à extraire, pour chaque objet détecté, des informations de forme, position, texture, etc...
3. **Mise en correspondance des caractéristiques** : On met en correspondance les objets détectés, en s'appuyant sur les caractéristiques, dans l'image courante avec les prédictions réalisées depuis l'image précédente.
4. **Prédiction** : On cherche à prédire la prochaine position des objets.

Le choix de notre algorithme de détection s'est assez vite porté sur un réseau de neurones. Des propositions hors apprentissage profond ont été étudiées, on peut par exemple citer la contribution de Dalal et al. [3], sur la détection de personnes par technique des Histogrammes de Gradient Orienté (HOG). Cependant, cette méthode est assez coûteuse en temps de calcul et nous craignons qu'elle n'ait pas le pouvoir de généralisation d'un réseau de neurones suffisamment alimenté en données.

Nous avons donc choisi le réseau de neurones Yolo dans sa troisième version [14] pour la détection des objets. Ce choix a été motivé par la haute performance du modèle en terme de temps de calcul, par rapport aux autres réseaux génériques de détection d'objets. On souhaite pouvoir proposer une solution qui puisse exploiter les ressources d'un ordinateur classique sans processeur graphique, pour limiter le coût du système. L'objectif est donc de minimiser le temps d'analyse d'une image, pour atteindre un rythme de traitement supérieur à 10 images par seconde. L'architecture sélectionnée se rapproche de YoloV3-tiny, une version

allégée de YoloV3 proposée par le framework Darknet². Comme expliqué précédemment, il convient de choisir un ensemble de boîtes d'ancrage (BA) adapté à notre jeu de données, on utilise donc un algorithme K-Means sur l'ensemble des boîtes annotées à la main pour obtenir six nouvelles BA que l'on renseigne dans les fichiers de configuration du réseau Yolo. On modifie également celui-ci pour qu'il ne détecte qu'une classe d'objets.

Au terme de l'entraînement on a un réseau capable de localiser les cochons sur les images, comme le montre la figure 7b.

Les étapes suivantes concernent le suivi des porcs entre les images. Pour assurer ce suivi, un porc qui apparaît dans l'image se voit associer un ensemble de coordonnées qui caractérisent sa Bounding Box (BB), ainsi qu'une identité propre et une vitesse. On prédit alors la position qu'aura ce cochon dans l'image suivante, à partir de sa position actuelle et de la vitesse déduite de sa position précédente.

À l'image suivante, on réalise à nouveau une détection de tous les cochons. On assigne ensuite les identités de tous les cochons suivis aux détections qui sont le plus proches des prédictions réalisées précédemment. Pour mesurer cette proximité, on utilise une distance artificielle construite en joignant l'*IoU* et la distance inter-centroïdes des deux boîtes comparées. Une assignation est réalisée à l'aide de l'algorithme Hongrois [8] pour trouver le meilleur appariement entre les prédictions et les détections.

Enfin, on obtient un visuel similaire à la sortie de la figure 9, où chaque cochon a sa boîte et son identifiant associés, et où le nombre en haut de l'image représente le nombre

2. <https://pjreddie.com/darknet/yolo/>

de cochons qui ont traversé la barre verticale au centre de l'image. Le schéma complet du système est présenté en figure 9.

S'agissant d'une étude préliminaire, aucune évaluation quantitative n'a pour le moment été implémentée. Seules quelques évaluations qualitatives, sur critère visuel, ont été réalisées sur une vidéo, pour qualifier les performances du système. Un élément important de l'évaluation sera la construction d'une base d'apprentissage conséquente et représentative des environnements dans lesquels les cochons seront suivis. Il sera nécessaire, comme pour l'application précédente, de développer un outil d'annotation dédié.

4 Conclusion

Cet article illustre le potentiel de récents travaux autour de la détection rapide d'objets par apprentissage profond, en particulier avec les réseaux Faster R-CNN et YOLO. Nous montrons également le potentiel de ces recherches dans un monde agricole en pleine mutation en abordant deux applications originales. Ces travaux soulignent également l'importance de disposer de données conséquentes et d'outils d'annotation adaptés pour permettre l'entraînement des réseaux profonds.

Références

- [1] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance : The clear mot metrics. *EURASIP Journal on Image and Video Processing*, 01 2008.
- [2] S. Chen, Y. Xu, X. Zhou, and F. Li. Deep learning for multiple object tracking : A survey. *IET Computer Vision*, 13, 01 2019.
- [3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 2, 06 2005.
- [4] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez. A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing*, 70 :41 – 65, 2018.
- [5] R. Girshick. Fast R-CNN. In *IEEE International Conference on Computer Vision*, pages 1440–1448, Washington, DC, USA, 2015. IEEE Computer Society.
- [6] I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.
- [7] M. Kaushal, B. Khehra, and A. Sharma. Soft computing based object detection and tracking approaches : State-of-the-art survey. *Applied Soft Computing*, 70, 05 2018.
- [8] Harold W. Kuhn. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, 2(1–2) :83–97, March 1955.
- [9] K. G. Liakos, P. Busato, D. Moshou and S. Pearson, and D. Bochtis. Machine learning in agriculture : A review. *Sensors*, 18(8), 2018.
- [10] T. Mengxiao, G. Hao, C. Hong, W. Qing, L. Chengjiang, and M. Yuhao. Automated pig counting using deep learning. *Computers and Electronics in Agriculture*, 163 :104840, 2019.
- [11] E. Psota, M. Mittek, L. C. Pérez, T. Schmidt, and B. Mote. Multi-pig part detection and association with a fully-convolutional network. In *Sensors*, 2019.
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once : Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, June 2016.
- [13] J. Redmon and A. Farhadi. Yolo9000 : Better, faster, stronger. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6517–6525, 2016.
- [14] J. Redmon and A. Farhadi. Yolov3 : An incremental improvement. *arXiv :1804.02767*, 2018.
- [15] S. Ruchti, G. Kratzer, R. Furrer, S. Hartnack, H. Würbel, and S. G. Gebhardt-Henrich. Progression and risk factors of pododermatitis in part-time group housed rabbit does in switzerland. *Preventive Veterinary Medicine*, 166 :56 – 64, 2019.

Vers un diagnostic des biais cognitifs par les opérateurs AGM

Valentin Fouillard¹

Safouan Taha²

Nicolas Sabouret¹

Frédéric Boulanger²

¹ Université Paris-Saclay, CNRS, LIMSI, 91400 Orsay France

² Université Paris-Saclay, CNRS, CentraleSupélec, LRI, 91405, Orsay, France

prenom.nom@universite-paris-saclay.fr

Résumé

De nombreux accidents impliquant une prise de décision par un opérateur humain montrent une incohérence entre les actions observées et les actions attendues. Dans cet article, nous proposons les premières pistes d'une explication de ce comportement à travers les biais cognitifs en se basant sur les propriétés de la révision de croyances AGM. Nous montrons qu'un opérateur respectant ces propriétés suffit à générer des comportements similaires aux biais sur un exemple. De plus nous montrons que caractériser un biais pourrait revenir à trouver une fonction de sélection parmi les comportements générés.

Mots Clef

Biais cognitif, AGM, révision de croyance, diagnostic, prise de décision

Abstract

Many accidents involving decision making by a human operator show an inconsistency between the expected and the observed actions. In this paper, we present an early stage model to explain this kind of behavior through cognitive biases, based on AGM belief revision properties. We show on an example that an operator that follows these properties can have biases-like behaviors. Moreover we show that characterizing a bias is similar to finding a selection function among the generated behaviors.

Keywords

Cognitive biases, AGM, belief revision, diagnostic, decision making

1 Introduction

Le 1er juin 2009, le vol 447 d'Air France entre Rio et Paris s'écrase dans l'océan atlantique. C'est l'accident le plus meurtrier d'Air France avec 228 morts. Trois ans plus tard, en juillet 2012, paraît le rapport du Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile [9]. L'analyse montre le déroulé suivant pour l'accident :

- incohérence temporaire entre les vitesses mesurées, vraisemblablement à la suite de l'obstruction des sondes Pitot par des cristaux de glace, ayant en-

traîné notamment la déconnexion du pilotage automatique ;

- situation de décrochage suite à la déconnexion du pilotage automatique ;
- maintien de la situation de décrochage par les pilotes par leurs actions.

De plus il peut être noté que l'alarme de décrochage a retenti plus de 75 fois sans que les pilotes identifient leur situation comme un décrochage. D'un point de vue extérieur et avec toutes les informations à notre disposition, ce comportement semble irrationnel. Pourtant si nous nous posons du point de vue du pilote avec les informations à sa disposition, ses actions semblent faire sens. En effet, le rapport montre que :

- l'alarme de décrochage sonnait de manière furtive et pouvait être considérée comme non pertinente ;
- il n'y avait pas d'information visuelle permettant de confirmer l'approche du décrochage après la perte des vitesses caractéristiques ;
- il y avait confusion possible avec une situation de survitesse ;
- des indications des directeurs de vol pouvaient conforter l'équipage dans ses actions, bien qu'elles soient inappropriées.

Nous voyons ici que nous avons affaire à une erreur de jugement de la situation qui diffère de la réalité. Cette erreur est confortée par les informations à la disposition des pilotes pendant la situation. Ce phénomène est largement étudié en psychologie dans le domaine des **biais cognitifs** et est présent jusque dans la méthode scientifique [22], la justice [15] ou en médecine [25]. Ainsi, comme le déclare Murata [21], si les ingénieurs oublient de prendre en compte et de comprendre les limitations humaines dans la conception des technologies modernes, la distorsion des décisions se produit inévitablement, aggravant les erreurs humaines et conduisant finalement à des incidents, des accidents, des collisions ou des catastrophes.

En informatique, la branche qui s'intéresse à modéliser et comprendre les erreurs dans les systèmes est le diagnostic [7]. Dans cet article, nous nous intéressons à un cas particulier de diagnostic. Il s'agit, à partir d'un comportement attendu et d'un comportement observé (suite d'actions)

qui semble irrationnel, d'apporter une explication en s'appuyant sur les biais cognitifs. Nous proposons pour cela de nous appuyer sur les modèles de révision de croyances pour déterminer quels biais peuvent être mis en œuvre pour mener à une erreur.

Dans la prochaine section, nous présentons la définition des biais cognitifs en sciences humaines et leur modélisation en informatique. Nous faisons le lien avec la rationalité telle qu'elle est représentée dans les modèles de révision de croyance. Dans la troisième section, nous montrons comment extraire des biais cognitifs à partir d'un ensemble de connaissances ayant conduit à une situation d'erreur. Nous illustrons ce fonctionnement sur un cas d'école. Nous montrons enfin dans les perspectives, quelles sont les propriétés que devrait respecter une fonction de sélection de biais dans un tel modèle.

2 Travaux connexes

Dans cette première section nous définissons les biais cognitifs selon les sciences humaines et discutons des modèles informatiques de ce phénomène irrationnel. Nous détaillons ensuite la raison pour laquelle la révision de croyance est le mécanisme central de la rationalité, et son lien avec le diagnostic.

2.1 Les biais cognitifs

Selon Tversky et Kahneman [33], l'être humain utilise des raccourcis de pensée (heuristiques) afin de compenser ses limitations en mémoire et en temps. Ces heuristiques peuvent parfois mener à des erreurs, c'est-à-dire des comportements qui diffèrent de la théorie du choix rationnel. Les auteurs montrent par exemple que les individus fondent leur jugement sur des informations personnelles plutôt que statistiques. Todd [31] montre l'importance de ces heuristiques comme par exemple l'heuristique *Take the Best*. Dans le problème de choisir une option parmi un ensemble de possibilités selon un critère de comparaison, *Take the Best* revient à prendre la première information qui départage les possibilités avec ces informations rangées par ordre de validité. La validité est la probabilité conditionnelle qu'un objet tombe dans une catégorie selon l'information choisie. Todd montre que *Take the best* est plus performant en généralisation et utilise moins d'information pour départager qu'une régression multiple. Mais dans certains cas, la solution obtenue grâce à cette heuristique est sous-optimale : nous parlons alors de *biais cognitif*.

Plusieurs études ont montré le rôle important de certains de ces biais dans la prise de décisions à forts risques. Par exemple, dans le domaine de l'aviation, Walmsley et Gibley [35] effectuent plusieurs expérimentations dans lesquelles un pilote est informé de la météo prévue en amont, puis un changement du temps apparaît pendant son trajet. Les différentes expériences montrent que :

- le bulletin météo donné avant l'envol influence l'interprétation des indices météorologiques par les pilotes (*le biais d'ancrage*);

- il n'existe pas de preuve montrant que les pilotes favorisent une stratégie de recherche d'éléments contradictoires par rapport à une stratégie de recherche de confirmation (*biais de confirmation*);
- les pilotes évaluent positivement le fait d'avoir traversé une zone de conditions météorologiques instables si le résultat est positif (*outcome bias*).

Le biais d'ancrage est le fait de s'appuyer sur l'information reçue en premier pour prendre une décision. *Le biais de confirmation* consiste à privilégier ses idées préconçues. Enfin l'*outcome bias* consiste à évaluer trop vite une décision quand son effet est connu. De plus, Gibley et Hill [14] effectuent des expérimentations similaires mais sur le scénario d'un pilote perdu qui doit retrouver son chemin. Une majorité des pilotes utilisent une stratégie de confirmation et prennent la mauvaise direction, mettant en avant ici encore le biais de confirmation. Enfin, nous pouvons citer Brafman & Brafman [6] qui montrent que *l'aversion à la perte* a largement contribué au crash du vol KLM 4805, l'aversion à la perte consistant à attacher plus d'importance à une perte qu'à un gain de même montant.

Dans l'industrie nucléaire, Takano et Reason [30] se sont basés sur des analyses des facteurs humains concernant les accidents dans des centrales nucléaires aux États-Unis, au Japon et sur un simulateur. Leur étude montre le rôle de l'excès de confiance, l'effet de récence qui est la facilité de se rappeler du dernier élément entendu et le biais de confirmation dans l'évaluation d'une situation inattendue.

Enfin, Murata [21] dans le domaine de l'accidentologie, met en évidence l'importance de certains biais cognitifs, notamment l'excès de confiance, le biais d'optimisme qui consiste à croire que les événements négatifs nous touchent moins que les autres, et le framing effect qui est le fait de choisir une option selon le fait quelle soit présentée positivement ou négativement.

L'ensemble de ces études permet d'identifier des biais fréquents dans la prise de décision : biais de confirmation, effet de récence, aversion à la perte, excès de confiance, etc. Nous nous concentrons sur ces biais pour construire des diagnostics sur les comportements erronés des opérateurs humains.

2.2 Modèles informatiques des biais

La littérature en informatique montre peu de travaux portant sur la modélisation des biais dans la prise de décision. La plupart des travaux en informatique décisionnelle visent à calculer une solution optimale [32] tandis que les travaux en simulation cherchent souvent à retrouver des faits stylisés [8].

Cependant, nous pouvons citer Voison [34] qui s'intéresse à la modélisation de l'impact des biais cognitifs dans les campagnes de vaccination. Ce modèle est basé sur un modèle de croyance et deux fonctions distinctes représentant chacune des biais. Le modèle mathématique est simple mais ne modélise que deux biais cognitifs (le biais de confirmation et de conformisme). Nous visons ici à avoir

une approche plus générale et à proposer un modèle prenant en compte plusieurs biais.

Le travail de Kulick [18] cherche à modéliser l'effet d'une opération militaire sur une cible. Ce modèle se base sur un modèle de « boîte noire » avec des facteurs en entrée pour aboutir à des probabilités de comportements possibles. Néanmoins, celui-ci n'offre pas la possibilité d'intégrer des mécanismes de biais connus, et du fait de son fonctionnement, le modèle ne donne pas le mécanisme mental qui mène au comportement en sortie.

De plus nous pouvons citer Arnaud et Adam [3] qui cherchent à modéliser les biais cognitifs lors des feux de forêts en se basant sur le modèle BDI [28]. Leur approche s'appuie sur un mécanisme de mise à jour de croyances en fonction de probabilités. Cette approche a l'avantage de pouvoir intégrer facilement de nouveaux biais dans le modèle et d'avoir une description claire du mécanisme mental effectué. Cependant, le choix d'une révision probabiliste des croyances de l'agent n'est pas conforme aux modèles de révision de croyances qui ont été largement étudiés et formalisés en logique.

Enfin Novaes et Veluwenkamp [24] étudient empiriquement deux modélisations du raisonnement humain. L'une se basant sur une logique non-monotone et l'autre sur une variation du cadre AGM appelée *screened revision*. Les auteurs mettent en avant que la deuxième modélisation est plus adéquate pour prédire le raisonnement humain, et plus précisément pour représenter le biais de confirmation. Dans cet article, nous utilisons la même variation d'AGM mais selon une approche différente. Nous partons d'une situation où le comportement est irrationnel et cherchons dans l'ensemble des révisions de croyances possibles si plusieurs biais peuvent être capturés, là où Novaes et Veluwenkamp cherchent à trouver le modèle qui prédit au mieux le biais de confirmation. Ainsi nous nous différencions par l'approche explicative plus que prédictive et par le nombre de biais possiblement capturés.

Nous pensons qu'en nous appuyant sur un mécanisme de révision de croyances et un modèle de type BDI, nous pourrions bénéficier d'une base théorique forte tout en offrant une facilité d'intégration à des modèles existants.

2.3 Révision de croyance

La révision des croyances est le mécanisme qui permet de passer d'un état de croyance à un autre tout en gardant la rationalité de l'agent. Ce dernier a été largement étudié en informatique [13].

Pour pouvoir étudier les biais cognitifs, nous proposons de nous appuyer sur ces modèles pour construire des transitions rationnelles entre des états pouvant expliquer un comportement irrationnel. Il faut donc que chaque étape soit conforme au modèle de révision de croyance, tout en menant à une situation d'erreur de décision (par exemple, le maintien de l'appareil en situation de décrochage malgré les alarmes lors du crash Rio-Paris).

Le problème de la révision de croyance est de choisir les

modifications à effectuer dans une base de connaissances pour prendre en compte une nouvelle information. Pour illustrer ce problème, nous prenons l'exemple proposé par Gärdenfors *et al.* dans leur livre fondateur [13]. Considérons la base de croyances suivante :

- (1) L'oiseau attrapé dans la cage est un cygne (A)
- (2) L'oiseau attrapé dans la cage vient de Suède (B')
- (3) La Suède fait partie de l'Europe ($B' \rightarrow B$)
- (4) Les cygnes européens sont blancs ($A \wedge B \rightarrow C$)

Nous pouvons en déduire alors que l'oiseau attrapé dans la cage est blanc (C). Imaginons maintenant que nous observons que l'oiseau est noir ($\neg C$). Nous devons alors réviser nos croyances afin de garder une cohérence dans notre base. Pour cela, nous devons éliminer une des croyances (1-4) mais d'un point de vue logique aucune préférence n'est donnée pour l'élimination : tous les sous ensembles de la base de croyance complétée par $\neg C$ sont acceptables pour réviser notre croyance

Bien que d'un point de vue logique l'élimination de toutes les anciennes croyances pour prendre en compte la nouvelle information soit recevable, elle n'est pas rationnelle. C'est pourquoi Alchourrón, Gärdenfors et Makinson (AGM) [1] définissent des postulats veillant à la rationalité de la révision des croyances. Le but de ces propriétés est de retenir autant que possible nos anciennes croyances, en d'autres termes, nous ne voulons changer que le nécessaire : c'est le *changement minimal*.

Le modèle AGM définit trois opérations possibles pour le changement de croyances : la contraction (\div), l'expansion ($+$) et la révision ($*$). La contraction est le fait de retirer une croyance de la base. L'expansion consiste à ajouter une nouvelle croyance à la base et la révision est l'ajout d'une nouvelle connaissance dans la base mais tout en gardant sa cohérence. Ainsi la révision peut être construite à partir de la contraction et de l'expansion selon l'*identité de Levi* [1] :

$$K * p = (K \div \neg p) + p$$

où K est la base de croyance et p une nouvelle information.

La **contraction** est donc au cœur de cette question de révision de croyances, et le modèle AGM caractérise ces fonctions de contractions rationnelles, appelées *partial meet contraction*. Ces travaux ont donné lieu à de nombreuses recherches [11] ainsi que plusieurs définitions de la révisions de croyances, dont il a été montré qu'elles sont équivalentes. En particulier, nous pouvons citer le *kernel contraction* [16] dont le principe est de calculer les ensembles minimaux impliquant la proposition à retirer de la base, et le *Epistemic Entrenchement* [12] qui définit une relation d'ordre sur les croyances de la base K pour abandonner en priorité les croyances avec un enracinement plus faible, c'est à dire celles qui ont un pouvoir explicatif faible. Enfin il existe certaines variations des opérateurs AGM permettant de prendre en compte les changements *non prioritaires*. C'est à dire que la nouvelle connaissance p n'est pas forcément ajoutée dans la base de croyance lors de la révision de croyance. Cette variation est appelée

screened revision. Enfin nous pouvons noter une extension du cadre AGM appelé *update* ou *mise à jour*. En effet deux cas peut être différenciés [17], soit la nouvelle information à prendre en compte est à propos du monde, soit le monde a changé. La logique de la mise à jour et de la révision sont différentes et ne retournent pas le même ensemble de croyances. La différence entre les deux se fait par le temps. Dans cet article nous nous limitons à des connaissances sur le monde au moment présent. C'est pourquoi nous utilisons la révision de croyance.

Dans notre article, nous utiliserons la méthode des Minimal Correction Set (MCS) qui consiste à calculer l'ensemble des corrections minimales à apporter à un ensemble de croyances insatisfiable pour le rendre satisfiable. Besnard [4] montre que MCS est le complément de *Maximal Subset Satisfiable*, ce qui revient à faire une *partial meet contraction* qui respecte donc AGM. Plus formellement :

$M \subseteq K$ est un MCS de K ssi :

- $K \setminus M$ est SAT
- $\forall p \in M, (K \setminus M) \cup \{p\}$ est UNSAT

De plus nous capturons le comportement de *screened revision* car il existera toujours le MCS contenant la nouvelle connaissance.

Il existe dans la littérature de nombreux algorithmes capables de calculer un tel ensemble [20].

Lien entre le diagnostic et la révision des croyances. Wassermann [36] montre qu'un problème de diagnostic peut être traduit en un problème de révision de croyance. En effet, l'approche *consistency-based diagnosis* proposée par [29] consiste à décrire un système en termes de composants et des interactions entre ces derniers indépendamment de la tâche de diagnostic. Le but est alors de trouver le composant responsable lorsqu'un comportement anormal est détecté. L'approche consistency-based diagnosis est un système de trois ensembles logiques :

- SD qui est la description du système ;
- ASS qui sont les hypothèses ;
- OBS qui sont les observations.

L'ensemble $\Delta \subset ASS$ est un diagnostic pour le triplet (SD, ASS, OBS) ssi Δ est un ensemble minimal tel que $SD \cup OBS \cup (ASS \setminus \Delta)$ est consistant. Ainsi, si nous partons de l'hypothèse que tous les composants sont fonctionnels alors le triplet sera inconsistant et Δ revient à trouver et retirer le (ou les) composant(s) défectueux. Wassermann montre que nous pouvons utiliser AGM pour calculer Δ car cela revient à trouver une partial meet contraction, c'est à dire trouver l'ensemble maximal satisfaisable.

Un tel diagnostic est donc l'ensemble minimal de propositions à retirer pour retrouver la consistence. Notre problème de diagnostic d'accident consiste, à partir d'une situation donnée et d'une prise de décision irrationnelle, à déterminer quelle croyance a été ignorée (retirée) par l'agent à cause d'un biais cognitif. Nous pouvons donc nous reposer sur l'opérateur de révision de croyance pour déterminer les croyances inconsistantes.

Dans la suite de cet article, nous montrons comment cette

approche de diagnostic à partir de révision de croyances peut être mise en œuvre pour extraire des biais cognitifs et expliquer des situations d'accident.

3 Contribution

Notre modèle part d'une situation où l'agent se comporte de manière irrationnelle. C'est à dire que certaines de ses actions sont incohérentes avec le monde. Le modèle est composé de deux ensembles, les croyances consistantes B de l'agent et ses observations Obs à un moment t . Nous partons de l'hypothèse que nous connaissons ses croyances et l'ensemble des observations que l'agent a pu faire au cours du temps. Pour trouver une explication possible de son comportement, nous cherchons à construire un état de croyance cohérente avec l'action au moment t en se basant sur B et Obs . De ce fait, nous traduisons ce problème de diagnostic en un problème de révision de croyances en se basant sur l'opérateur AGM. Pour cela nous considérons le système $\{B \cup Obs \cup Action\}$ et cherchons les MCS sur cet ensemble. Deux cas sont alors possible :

1. L'ensemble des MCS est vide, ce qui veut dire que l'agent n'a pas eu besoin de faire d'ignorance dans ses croyances pour effectuer l'action.
2. Sinon chaque MCS correspond à une révision de croyance, cohérente avec l'action.

Nous allons montrer à travers un exemple que les MCS permettent d'avoir un ensemble comportant des biais cognitifs possibles (tout en préservant la rationalité de l'agent) et que caractériser un biais spécifique reviendrait à sélectionner la solution représentant ce biais. En d'autres termes, nous montrons ici les premiers indices que la révision de croyances est suffisante pour mettre en évidence des biais cognitifs. De plus, nous discuterons des caractéristiques nécessaires des fonctions de sélection de biais.

3.1 Exemple concret

Nous allons considérer le problème suivant, inspiré de l'expérimentation de Walmsley et Gibley [35] présentée dans la section précédente. À l'approche de son point d'arrivée, un pilote d'avion doit prendre en compte de nouvelles informations météorologiques moins favorables. Nous pouvons modéliser ce problème avec la base de croyances suivante :

- (1) $(visibility \leq 16) \rightarrow \neg good$
Si la visibilité est en dessous de 16 km alors les conditions de vol sont mauvaises
- (2) $(cloud \leq 2500) \rightarrow \neg good$
Si les nuages sont en dessous de 2500 pieds alors les conditions de vol sont mauvaises
- (3) $meteo : ((visibility = 25) \wedge (cloud = 3800))$
La météo fournie au début du vol déclare une visibilité de 25km et des nuages à 3800 pieds
- (4) $DoneTraj \rightarrow good$
Je ne peux faire le trajet que si les conditions sont bonnes

et l'observation suivante :

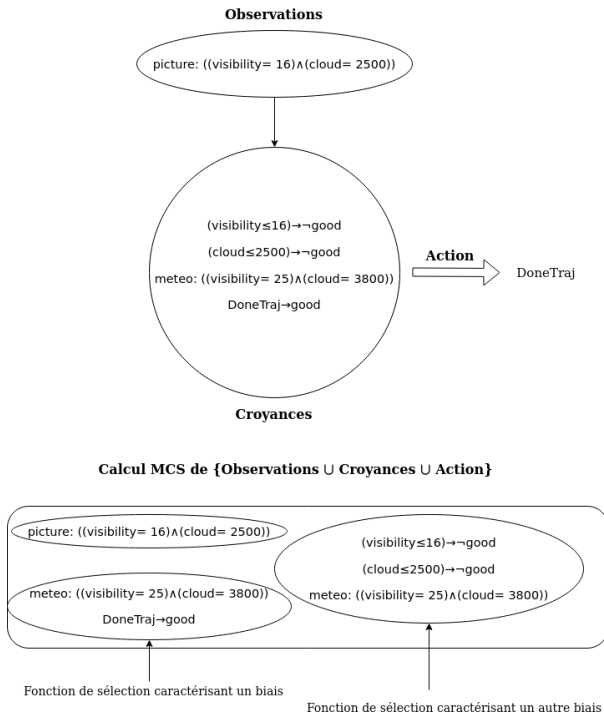


FIGURE 1 – Modèle du diagnostic des biais cognitifs

(5) $picture : ((visibility = 16) \wedge (cloud = 2500))$

La photographie reçue à l’approche du point d’arrivée montre une visibilité de 16 km et des nuages à 2500 pieds

Nous nous plaçons dans la situation où le pilote effectue le trajet (décision erronée) : *DoneTraj* est nécessairement vrai. Le système à résoudre est alors la base de croyances [1-4], l’observation *picture* (5) et l’action *DoneTraj*. Le système est d’entrée de jeu insatisfaisable car l’agent ne peut pas accepter plusieurs valeurs différentes pour la visibilité et les nuages (propositions 3 et 5). De plus si l’agent prend en compte les nouvelles informations qui lui ont été communiquées sur la photographie, le système reste insatisfaisable, du fait que *DoneTraj* doit être vrai. Nous allons donc utiliser un algorithme de MCS pour calculer les corrections rationnelles possibles pour rendre cet ensemble satisfaisable. Chaque correction correspond à une révision de croyance possible pouvant expliquer *DoneTraj*. Cela correspond à l’ensemble des croyances que l’agent a dû ignorer pour effectuer cette action. Dans notre exemple, nous nous retrouvons avec l’ensemble de solutions suivantes :

$$\{\{3, 1, 2\}, \{3, 4\}, \{5\}\}$$

3.2 Extraction de biais cognitifs

Dans cette partie, nous allons discuter de chacune de ces solutions :

- $\{5\}$ représente le fait d’ignorer (que ce soit involontairement ou volontairement) les informations météorologiques contenues dans la photo.
- $\{3, 1, 2\}$ représente le fait d’ignorer les connaissances sur le danger et d’ignorer (involontairement ou volontairement) les informations données par la météo au début du vol.
- $\{3, 4\}$ représente le fait de considérer le danger comme négligeable pour faire le trajet. L’agent ignore ici encore les informations météorologiques initiales (pour garder une seule valeur possible pour les nuages et la visibilité).

Si nous gardons les hypothèses de l’étude où les informations sur les photos et les données météorologiques ne pouvaient être ignorées involontairement, alors le cas $\{5\}$ vient capturer le comportement d’un biais d’ancrage mis en évidence dans l’étude. En effet, ignorer ces informations revient à placer trop d’importance dans la véracité des données météorologiques et oblige à ignorer les informations de la photo : “Specifically, pilots appeared to place too much importance on reports that may no longer be valid and then fail to adjust their perceptions in the face of evidence to the contrary (i.e. the presented images)” [35].

Ignorer la météo (3), c’est à dire donner une priorité de croyance sur la photographie, revient à faire deux choix de révision possibles. Soit l’agent ignore ses croyances sur les conditions de danger ($\{3, 1, 2\}$), soit il ignore le fait qu’il est nécessaire pour ce trajet que les conditions soient bonnes ($\{3, 4\}$). Bien que ces comportements ne ressortent pas dans l’étude de Walmsley et Gilbey, ils n’en restent pas moins intéressants. Par exemple ignorer $\{3, 4\}$ peut être vu comme un excès de confiance : l’agent ne remet en cause ni les photographies ni le danger, mais considère que, dans son cas, les conditions pour faire le trajet sont suffisantes. Enfin $\{3, 1, 2\}$ peut correspondre à une aversion à la perte, c’est-à-dire que le fait de ne pas continuer le trajet est une perte trop importante aux yeux du pilote et qu’il est préférable d’ignorer le danger et de continuer. Néanmoins, il faut garder en mémoire que les solutions du MCS ne représentent pas uniquement des ignorances volontaires : il peut aussi s’agir d’oublis involontaires et donc de choix non-intentionnels de la part de l’agent.

Enfin, nous avons considéré dans toutes les hypothèses mentionnées ci-avant que l’agent percevait toutes les informations (ce qui était le cas dans l’étude de Walmsley et Gilbey).

3.3 Sélection d'une solution

Chaque MCS correspond donc à une solution d'explication du comportement irrationnel par un raisonnement rationnel. Nous considérons donc que les biais sont des erreurs de fonction de sélection dans l'ensemble des révisions de croyances possibles. Néanmoins, si nous voulons définir quelle solution, parmi l'ensemble des possibilités, correspond à un biais en particulier, nous devons prendre en compte des caractéristiques qui ne sont pas présentes dans la modélisation de l'exemple. Par exemple, quels facteurs ont influencé le pilote pour prendre la décision d'ignorer les informations de la photographie plutôt que de prendre la décision d'ignorer la météo et le danger ? En d'autres termes, pourquoi la météo est-elle difficile à abandonner dans le premier cas et les informations dans la photographie dans le deuxième cas ? Ces points seront adressés dans de futurs travaux dont nous donnons les premières pistes dans la section suivante.

4 Travaux Futurs

Nous nous retrouvons avec un ensemble de solutions de révision dont le comportement de sortie peut correspondre à des biais. Néanmoins, nous n'avons pas défini une fonction de sélection permettant de dire quelle croyance ignorée correspond à un biais. Si nous nous référons aux résultats de l'étude [35], une majorité de pilotes a choisi d'ignorer {5} et non {3, 1, 2} ou {3, 4}. De ce fait, il faut que la fonction de sélection soit une fonction de préférence sur l'ensemble des solutions. Or l'*epistemic entrenchement* [12] a ce fonctionnement comme vu à la section 2.3. Néanmoins il faut pouvoir définir les facteurs de préférence d'une croyance, et le fonctionnement des biais cognitifs nous donne quelques pistes. En effet, le biais d'ancrage donne une importance plus grande à la première information reçue (ici, 3 a plus d'importance que 4). Une notion de temps doit donc être ajoutée dans la logique afin de prendre en compte ce facteur. De plus, le temps n'est pas la seule notion importante si nous nous intéressons à d'autres biais. Par exemple le biais de confirmation qui est très proche de l'ancrage vient ignorer les informations qui rentrent en contradiction avec nos croyances de base. Comme le décrit Nickerson [23], une croyance ancienne A et appuyée par d'autres croyances B ($B \rightarrow A$) sera difficile à abandonner et sera sujette à un biais de confirmation. Là encore cette notion d'enracinement doit être prise en compte dans la sélection. Enfin les émotions jouent un rôle aussi dans les biais cognitifs. Par exemple le biais *attentional* [19], nous dirige vers des croyances qui ont un sens émotionnel. Par exemple un pilote ayant une expérience traumatisante d'une panne d'essence fera tout pour éviter de retomber dans le même contexte (et donc possiblement prendra d'autres risques). De plus, un biais n'est pas limité à un seul facteur mais peut aussi dépendre d'une combinaison de ceux-ci. Par exemple, English et Soder [10] montrent que le biais d'ancrage est plus prononcé lorsque nous sommes tristes.

Ainsi le travail futur reposera essentiellement sur la définition de fonctions de sélection correspondant à des biais spécifiques. Ces fonctions reposeront sur des facteurs d'évaluation avec notamment :

- le temps
- les émotions
- l'enracinement des croyances

De ce fait, ces fonctions permettront de mettre en évidence, dans une situation donnée, si un biais cognitif a pu jouer un rôle ou non. De plus nous voyons que plusieurs biais cognitifs prennent en compte une combinaison de tous ces facteurs et que nous devons trouver un moyen d'agencer tous ces éléments pour caractériser un biais. En s'appuyant sur la littérature des sciences humaines, une étude plus poussée des différents facteurs à prendre en compte dans le fonctionnement des biais cognitifs permettra sûrement de mettre en valeur d'autres facteurs mais aussi l'importance de ces facteurs. Par exemple nous pouvons voir que la notion de temps est centrale pour la conservation des croyances.

Diagnostic ou prédiction ? Notre modèle se basant sur la révision de croyance a l'avantage d'analyser une situation passée (section 2.3), mais peut elle aussi prédire et corriger des biais ? La révision de croyance par un opérateur AGM nous donne un ensemble de monde possible où le changement est minimal. Néanmoins aucune indication n'est donnée pour choisir le monde préférable (et donc le monde le plus probable pour l'agent). Nos fonctions de sélection pourront indiquer si le choix d'un monde correspond à un certain biais, sans pour autant fournir aucune information sur une possible préférence dans le choix. Il n'est donc pas possible de faire de la prédiction.

Cependant, en enrichissant la base de connaissance avec des aspects métiers, il est envisageable de prioriser les différents biais à partir des paramètres des fonctions de sélection et d'un « coût » des altérations de la base de connaissance lors de la révision de croyances. Là encore nous laissons cette piste de réflexion pour de futur travaux.

5 Conclusion

Dans cet article, nous avons proposé un modèle général pour diagnostiquer les biais cognitifs dans un modèle logique d'action, comme le modèle BDI [28] ou le calcul des situations [27]. Notre modèle a l'avantage de reposer sur la théorie de la révision de croyances qui a déjà été largement étudiée et formalisée dans la littérature. De plus, nous montrons que la génération des comportements possibles par la révision de croyances est suffisante pour retrouver des comportements similaires aux biais cognitifs tout en préservant la rationalité de la révision de croyances de l'agent. La prochaine étape de nos travaux est de caractériser la fonction de sélection venant choisir le comportement relatif à un biais spécifique. Nous avons donné dans cet article quelques premières pistes sur les facteurs qu'il est nécessaire d'intégrer pour déterminer un biais cognitif. Ce dernier point fera l'objet d'un travail futur pour spécifier quels

facteurs ont une influence sur quels types de biais. Nous pouvons pour ce point nous appuyer sur la littérature des sciences humaines. Pour l'intégration, la littérature de la révision de croyances étant déjà très fournie, de nombreux travaux se sont déjà penchés sur le problème du temps [5], des émotions [26] et de l'enracinement des croyances [12]. Enfin Arnaud et Adam [2] ont mis en évidence les limitations du modèle BDI pour rendre compte des comportements humains. Notamment dans l'engagement dans les intentions, la mise à jour des croyances et le changement de contexte. Les auteurs mettent en évidence l'importance de prendre en compte les sciences sociales et d'intégrer les biais cognitifs dans la BDI. Nous pensons que les prémisses de notre travail offrent un point de départ pour répondre à cette problématique.

Références

- [1] Carlos E Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change : Partial meet contraction and revision functions. *The journal of symbolic logic*, 50(2) :510–530, 1985.
- [2] Maël Arnaud, Carole Adam, and Julie Dugdale. Les limites du BDI pour rendre compte du comportement humain en situation de crise. In *Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA 2017)*, Caen, France, July 2017.
- [3] Maël Arnaud, Carole Adam, and Julie Dugdale. The role of cognitive biases in reactions to bushfires. In *ISCRAM*, Albi, France, May 2017.
- [4] Philippe Besnard, Éric Grégoire, and Jean-Marie Lagniez. On computing maximal subsets of clauses that must be satisfiable with possibly mutually-contradictory assumptive contexts. In *AAAI*, 2015.
- [5] Giacomo Bonanno. Axiomatic characterization of the agm theory of belief revision in a temporal logic. *Artificial Intelligence*, 171(2-3) :144–160, 2007.
- [6] Ori Brafman and Rom Brafman. *Sway : The irresistible pull of irrational behavior*. Crown Business, 2008.
- [7] Marie-Odile Cordier, Philippe Dague, Yannick Pencolé, and Louise Travé-Massuyès. Diagnostic et supervision : approches à base de modèles. In Pierre Marquis, Odile Papini, and Henri Prade, editors, *Panorama de l'intelligence artificielle : Ses bases méthodologiques, ses développements*, volume 2. Cépaduès, January 2013.
- [8] Paul Davidsson. Agent based social simulation : A computer science view. *Journal of artificial societies and social simulation*, 5(1), 2002.
- [9] Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile. Bea f-cp090601. Technical report, Ministère de l'Écologie, du Développement durable et de l'Énergie, 2012.
- [10] Birte Englich and Kirsten Soder. Moody experts—how mood and expertise influence judgmental anchoring. *Judgment and Decision making*, 4(1) :41, 2009.
- [11] Eduardo L. Fermé and Sven Ove Hansson. Agm 25 years. *Journal of Philosophical Logic*, 40 :295–331, 2011.
- [12] Peter Gärdenfors. Epistemic importance and minimal changes of belief. *Australasian Journal of Philosophy*, 62(2) :136–157, 1984.
- [13] Peter Gärdenfors, Hans Rott, DM Gabbay, CJ Hogger, and JA Robinson. Belief revision. *Computational Complexity*, 63(6), 1995.
- [14] Andrew Gilbey and Stephen Hill. Confirmation bias in general aviation lost procedures. *Applied Cognitive Psychology*, 26(5) :785–795, 2012.
- [15] Julien Goldszlagier. L'effet d'ancrage ou l'apport de la psychologie cognitive à l'étude de la décision judiciaire. *Les Cahiers de la Justice*, 2015(4) :507–531, 2015.
- [16] Sven Ove Hansson. Kernel contraction. *The Journal of Symbolic Logic*, 59(3) :845–859, 1994.
- [17] Hirofumi Katsuno and Alberto O. Mendelzon. On the difference between updating a knowledge base and revising it. In Peter Gärdenfors, editor, *Belief Revision*, Cambridge Tracts in Theoretical Computer Science, page 183–203. Cambridge University Press, 1992.
- [18] Jonathan Kulick and Paul K Davis. Modeling adversaries and related cognitive biases. *Modeling Adversaries and Related Cognitive Biases*, 2003.
- [19] Colin MacLeod, Andrew Mathews, and Philip Tata. Attentional bias in emotional disorders. *Journal of abnormal psychology*, 95(1) :15, 1986.
- [20] Joao Marques-Silva, Federico Heras, Mikolás Janota, Alessandro Previti, and Anton Belov. On computing minimal correction subsets. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.
- [21] Atsuo Murata, Tomoko Nakamura, and Waldemar Karwowski. Influence of cognitive biases in distorting decision making and leading to critical unfavorable incidents. *Safety*, 1(1) :44–58, 2015.
- [22] Clifford R Mynatt, Michael E Doherty, and Ryan D Tweney. Confirmation bias in a simulated research environment : An experimental study of scientific inference. *Quarterly Journal of Experimental Psychology*, 29(1) :85–95, 1977.
- [23] Raymond S Nickerson. Confirmation bias : A ubiquitous phenomenon in many guises. *Review of general psychology*, 2(2) :175–220, 1998.
- [24] Catarina Novaes and Herman Veluwenkamp. Reasoning biases, non-monotonic logics and belief revision. *Theoria*, 83, 12 2016.

- [25] ED O’Sullivan and SJ Schofield. Cognitive bias in clinical medicine. *JR Coll Physicians Edinb*, 48(3) :225–32, 2018.
- [26] César F Pimentel and Maria R Cravo. Affective revision. In *Portuguese Conference on Artificial Intelligence*, pages 115–126. Springer, 2005.
- [27] Javier Andres Pinto and Raymond Reiter. *Temporal reasoning in the situation calculus*. University of Toronto, 1994.
- [28] Anand S Rao, Michael P Georgeff, et al. Bdi agents : from theory to practice. In *ICMAS*, volume 95, pages 312–319, 1995.
- [29] Raymond Reiter. A theory of diagnosis from first principles. *Artificial intelligence*, 32(1) :57–95, 1987.
- [30] Kenichi Takano and James Reason. Psychological biases affecting human cognitive performance in dynamic operational environments. *Journal of Nuclear Science and Technology*, 36(11) :1041–1051, 1999.
- [31] Peter M Todd and Gerd Gigerenzer. Précis of simple heuristics that make us smart. *Behavioral and brain sciences*, 23(5) :727–741, 2000.
- [32] Alexis Tsoukiàs. From decision theory to decision aiding methodology. *European Journal of Operational Research*, 187(1) :138–161, 2008.
- [33] Amos Tversky and Daniel Kahneman. Judgment under uncertainty : Heuristics and biases. *science*, 185(4157) :1124–1131, 1974.
- [34] Marina Voinson, Sylvain Billiard, and Alexandra Alvergne. Beyond rational decision-making : modelling the influence of cognitive biases on the dynamics of vaccination coverage. *PLoS one*, 10(11), 2015.
- [35] Stephen Walmsley and Andrew Gilbey. Cognitive biases in visual pilots’ weather-related decision making. *Applied Cognitive Psychology*, 30(4) :532–543, 2016.
- [36] Renata Wassermann. An algorithm for belief revision. Technical report, In Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000), 2000.

IA et assistance à la décision humaine : influence des informations transmises sur l'évaluation d'un nouveau plan de vol.

Adrien Metge, Nicolas Maille

ONERA, BA 701, 13661, SALON cedex AIR

adrien.metge@onera.fr, nicolas.maille@onera.fr

Résumé

L'accroissement du niveau d'autonomie décisionnelle des systèmes de drones nécessite l'introduction d'algorithmes complexes qui permettent d'envisager un champ opérationnel plus large, mais dont le fonctionnement devient plus opaque pour l'opérateur en charge de la supervision. Cette étude s'appuie sur les mécanismes cognitifs sous-jacents à la prise de décision humaine pour définir des outils d'amélioration de la transparence des propositions du système. Une expérimentation en micro-monde est menée pour étudier l'influence de ces différents types d'assistance sur le jugement de l'opérateur dans le cadre d'une replanification de mission de drone.

Mots-clés

Intelligence artificielle explicable, interactions homme-machine, prise de décision

Abstract

Increasing the level of decision-making autonomy of UAV systems requires the introduction of complex algorithms that allow a wider operational field to be envisaged, but whose operation becomes more opaque for the operator in charge of supervision. This study is based on the cognitive mechanisms underlying human decision making to define tools to improve the transparency of the system propositions. A micro-world experiment is conducted to study the influence of these different types of assistance on the operator's judgement in the context of UAV mission replanning.

Keywords

Explainable artificial intelligence, human-agent interaction, decision making

1 Introduction

Le rôle des systèmes automatisés est un enjeu central pour l'évolution de nos sociétés [5]. Dans le domaine aéronautique, la mise en œuvre des systèmes de drones marque une étape dans la coopération homme-machine. La séparation géographique entre le vecteur aérien et la station au sol où se trouvent les opérateurs entraîne l'augmentation du niveau d'autonomie de l'appareil face à certains aléas, tels que des changements rapides de situation ou des pertes de communication [1]. Il est actuellement souhaité que l'homme reste au centre des décisions critiques et que la responsabilité de l'action lui soit allouée, que ce soit pour des raisons tactiques ou éthiques [13], [4]. La prise de décision humaine repose alors de plus en plus sur un prétraitement de l'information qui doit améliorer la compréhension que l'opérateur peut avoir de la situation autour de l'appareil et de son état de fonctionnement. L'essor des algorithmes d'intelligence artificielle et de leurs capacités de traitement peut être vu comme une opportunité pour aborder ces situations complexes, mais aussi comme un risque quant à la possibilité pour l'opérateur de garder son rôle de décideur. Pour répondre à cet enjeu des recherches sont menées sur le développement de fonctionnalités permettant à des agents artificiels d'expliquer leur comportement et se fédèrent autour du terme "intelligence artificielle explicable" [3]. Un algorithme est dit transparent dans la mesure où il peut donner à ses utilisateurs une vision claire des procédures employées et des fonctionnalités qu'il remplit, afin d'en permettre un usage averti [11].

Afin d'être réellement au service du décisionnaire et de lui apporter une plus-value, des outils de

recommandation ergonomiques doivent intégrer le cadre et les limitations cognitives, émotionnelles et sociales qui structurent la prise de décision humaine [8]. L'explication est un processus social qui consiste en un transfert de connaissance afin de bâtir une conscience de situation et une confiance appropriée, c'est à dire calibrée sur la fiabilité des différentes fonctionnalités du système [2]. On peut identifier plusieurs approches pour la prise de décision, qui permettent de créer différents types d'explications :

Approche normative. L'approche normative de la théorie de la prise de décision postule la rationalité de l'opérateur [14]. Celui-ci est supposé anticiper lors de chaque situation tous les événements possibles, et estimer leur probabilité de réalisation. En comparant l'utilité subjective qu'il tirerait de chaque issue, il peut ainsi déterminer la solution optimale à suivre. Dans cette approche, un système automatisé peut indiquer les coûts qu'il associe à chaque objet, pour que l'opérateur puisse faire les calculs avec sa propre échelle de valeurs.

Approche descriptive. L'approche descriptive de la théorie de la prise de décision observe qu'en situation naturaliste l'agent ne s'attend pas à recevoir toutes les explications ni une liste causale exhaustive [7]. Une bonne explication, pour pouvoir être le pilier d'une prise de décision, doit être sélectionnée de manière contextuelle pour son caractère contrastif, en regard d'autres éléments [12]. Il s'agit de permettre de mieux mesurer la différence entre ce qu'on attend et le résultat effectif, et d'interroger sur ce qu'on ne sait pas. Dans ce cadre, le système devrait mettre en exergue ce qui caractérise la solution proposée par rapport aux autres possibilités.

Approche constructive. L'approche constructive de la théorie de la prise de décision vise l'apparition d'un consensus à travers un dialogue sur les modèles du monde des décideurs [13]. Il s'agit de construire une représentation partagée entre les membres d'un groupe de décision en confrontant leurs points de vue. Pour cela, il faut identifier les spécificités réciproques de l'homme et de la machine dans les processus d'acquisition et de traitement de l'information, pour pouvoir tirer profit des forces de chacun et combler leurs faiblesses. Dans ce cadre le processus explicatif doit être itératif et bidirectionnel. Il accompagne l'émergence de la solution.

Dans l'étude présentée dans cet article, nous nous concentrons sur les situations où l'opérateur est amené à prendre une décision critique en étant

assisté par un système doté d'un algorithme d'intelligence artificielle. Le cas applicatif choisi est celui de la supervision d'un système de drone prenant en compte l'évolution d'une situation tactique pour replanifier sa mission au cours de sa réalisation. Nous cherchons plus particulièrement à évaluer comment la nature des informations sur les différentes options envisagées, transmises par le système à l'opérateur, modifie la prise de décision.

Nous présentons dans le paragraphe 2 l'expérimentation réalisée et les résultats obtenus, puis nous les discutons dans le paragraphe 3.

2 Etude expérimentale

Cette étude est la première brique d'un programme de recherche visant à optimiser les interactions entre un opérateur et un système fortement automatisé lors de la prise de décision critiques. Dans cette première phase nous nous concentrons sur les approches normatives et descriptives. Nous avons implémenté un micro-monde permettant d'étudier de manière contrôlée l'influence des types d'informations disponibles sur le processus de prise de décision d'un opérateur. Nous cherchons à évaluer si la nature des explications modifie le comportement de l'opérateur et sa prise de décision finale. L'algorithme de calcul émulé est basé sur un fonctionnement normatif, et les explications ajoutées de type descriptif. Nos hypothèses sont :

H1 : L'ajout d'explications améliore la prise de décision.

H2 : L'ajout d'explications améliore la confiance de l'opérateur dans la décision prise.

H3 : L'ajout d'explications ralentit la prise de décision.

2.1 Participants

Un ensemble de 12 personnes en bonne santé, docteurs et chercheurs en aéronautique (25% de femmes), d'âge moyen 28.9 ans (écart-type : 10.4 ans) a participé à cette étude. Aucun d'entre eux n'avait une connaissance préalable de la visée de l'expérience.

2.2 Description de la tâche

Les participants incarnaient un opérateur aérien militaire chargé de superviser un drone pour la réalisation de missions en territoire ennemi. L'objectif des missions était de survoler plusieurs

cibles afin de les photographier, puis de sortir de la zone ennemie, et ce en minimisant les risques pris. L'opérateur était assisté par une « intelligence artificielle » chargée à la fois de détecter et classifier pendant la mission les nouvelles menaces (traitement d'images), et de les prendre en compte pour déterminer le plan de vol le plus intéressant à emprunter grâce à un algorithme de recherche de chemin. Pour cela, le territoire ennemi était discrétisé en une grille dont chaque case possédait une valeur de risque qui prenait en compte plusieurs critères : altitude, dangerosité des ennemis à proximité, temps cumulé de vol (Figure 1).

Les missions se déroulaient sur des territoires différents mais avec un scénario similaire : 1) le drone se dirige vers la zone ennemie avec un plan de vol initial 2) soudain des entités ennemies sont détectées 3) alors le système calcule un nouveau plan de vol et le propose à l'opérateur 4) puis le drone arrive à l'entrée de la zone ennemie et attend que l'opérateur évalue sa proposition de plan (Figure 2). Cette évaluation comprenait deux parties : juger si le plan proposé par le système est le plan optimal ou pas, puis indiquer la confiance qu'a l'opérateur dans sa décision.

2.3 Construction des systèmes d'aide à la décision

Pour permettre à l'opérateur d'évaluer les propositions de plan de vol faites par le drone puis de faire un choix éclairé, nous avons développé trois outils d'aide à la décision renseignant sur les

caractéristiques de ces plans. Ces outils étaient consultables librement en cliquant sur des boutons affichés sur l'interface. De cette manière, l'échange entre l'opérateur et le système était consenti et volontaire [9]. L'outil A implémente des explications normatives, c'est-à-dire relatives à l'optimisation des coûts sur un chemin particulier, et les outils B et C des explications descriptives.

Outil A. Ce premier outil explicite pour l'opérateur les fonctions de coût utilisées par l'algorithme (partie légende de la carte) et le coût résultant associé au chemin proposé. Comme l'optimisation est multicritère, l'outil renseigne sur le risque objectif du plan de la machine selon les différentes dimensions. Ceci lui permet donc de se faire une idée sur les compromis altitude/danger/temps. Ce niveau se situe au niveau de la pure optimisation mathématique.

Outil B. Cet outil renseigne sur la non-exhaustivité des chemins calculés par la machine. La situation étant dynamique, le système n'a pas les ressources temporelles suffisantes pour simuler l'entière des chemins possibles. A travers l'utilisation d'heuristiques de calcul, il va sélectionner un nombre limité de chemins à évaluer. Il est donc possible qu'il n'en examine pas certains qui soient d'excellente qualité, ce dont l'opérateur pourrait se rendre compte. L'opérateur se voit donner la possibilité de regarder deux autres grands types de chemins déjà envisagés par le drone, mais plus risqués. Ce type d'explication se veut contrastif, en permettant d'explorer un « pourquoi pas » un autre choix.

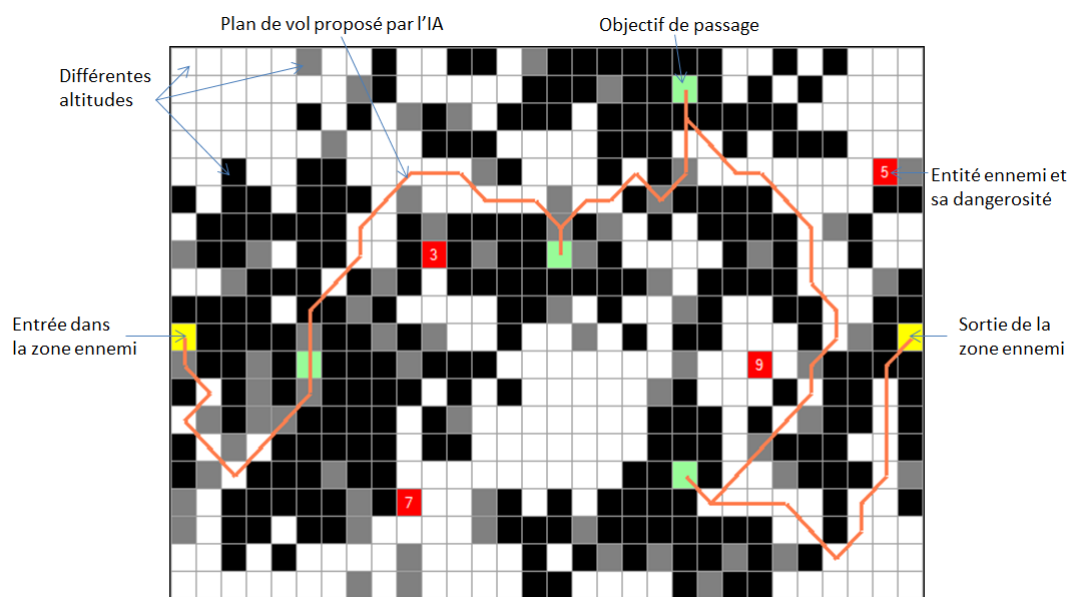


Figure 1 - Carte du territoire d'une mission et plan proposé par l'IA

Outil C. Cet outil ajoute des précisions sur le fonctionnement du module de vision artificielle qui identifie et classe les menaces potentielles, et qui était simulé dans l'expérimentation. On donne à l'opérateur la possibilité de voir l'image des objets classifiés par le système comme dangereux, ainsi que la confiance du système dans sa classification, représentée par une probabilité. En permettant de s'assurer de la validité des menaces, ce niveau rend possible la vérification des causes plus profondes de la replanification, ce qui est utilisé ensuite pour calculer les risques. En effet, les systèmes d'IA sont faillibles et connaissent un certain nombre de biais intrinsèques [10]. Il s'agissait, à travers ces explications, de permettre à l'opérateur de prendre conscience de limites potentielles dans le raisonnement de planification du système. Par comparaison des images et de la classification associée, l'opérateur pourrait détecter de telles erreurs, et inférer les modifications de planification qu'elles impliquent.

2.4 Plan expérimental

Nous avons mis en place un design expérimental cumulatif pour favoriser la complémentarité des différentes approches de la prise de décision, en faisant varier à la fois la quantité et la nature des explications des propositions. Cette étude comprenait ainsi trois conditions expérimentales :

Condition A : l'opérateur n'avait accès qu'à l'outil A pour prendre sa décision,

Condition B : l'opérateur pouvait se servir des outils A et B pour assoir sa décision,

Condition C : les trois outils A, B et C était utilisables par l'opérateur.

Tous les participants ont effectué 6 missions d'entraînement, puis 12 missions (33% dans chaque condition) dont le déroulement a été enregistré. Chaque participant s'est vu proposer par l'IA dans 66% des missions un plan optimal, et dans 33% des missions un plan sous-optimal. Ces proportions permettent de proposer un système performant la majorité du temps, ce qui justifie son utilisation, mais le fait que tous les plans ne soient pas optimaux permet de mesurer si les différents outils aident l'opérateur à identifier les solutions sous-optimales. L'enchaînement des missions et des conditions expérimentales a de plus été équilibré pour chaque participant et entre tous participants, afin de minimiser les potentiels effets de séries qui pourraient survenir.

2.5 Métriques

Un chronomètre qui démarrait à l'instant où le système proposait un nouveau plan était affiché sur l'interface. Le temps mis par l'opérateur pour prendre sa décision sur la qualité du plan proposé était enregistré et utilisé pour répondre à l'hypothèse

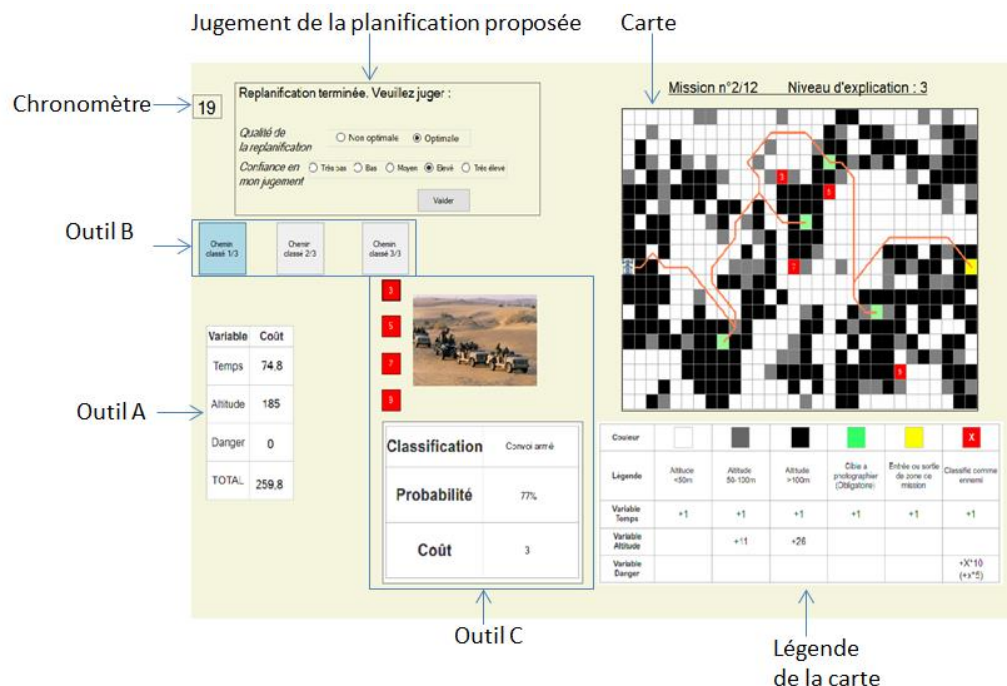


Figure 2 - IHM de supervision

H3. Pour chaque mission, les participants devaient indiquer s'ils pensaient que le plan proposé par l'IA était optimal ou non (mesure pour l'hypothèse H1), ainsi que leur confiance dans ce jugement sur une échelle allant de « très bas » à « très élevé » (mesure pour l'hypothèse H2). La pertinence de la réponse de l'utilisateur est évaluée en confrontant sa réponse sur l'optimalité du plan à la réalité.

De plus, afin d'objectiver la manière dont les participants avaient utilisé les aides proposées, l'ensemble des interactions avec l'interface (appuis sur les boutons liés à l'activation des outils B et C) étaient enregistrés. Enfin, après avoir réalisé toutes les missions, un entretien individuel permettait de recueillir des avis subjectifs sur la plus-value apportée par les différents systèmes d'aide à la décision.

2.6 Résultats

Les données des 12 participants sont incluses dans l'analyse. Des analyses de variances (ANOVA) inter-individus ont été réalisées, le seuil de significativité étant fixé à $p=0.05$. Tout d'abord, on n'observe pas d'influence des différents outils d'aide sur la pertinence de la réponse du participant ($F(2,141) = .30$, $p = .74$), ni sur sa confiance en son jugement ($F(2,141) = .46$, $p = .62$). Les hypothèses H1 et H2 ne sont donc pas validées dans ce cadre expérimental. Néanmoins, selon le type d'explication associé à chaque mission, le temps de réponse du participant varie significativement ($F(2,141) = 10.51$, $P < .001$) (Figure 3). Cela renforce l'hypothèse H3. Une analyse post-hoc grâce à un test de Tukey montre des différences significatives entre le temps mis pour prendre la décision entre les conditions A et B ($p = .019$), ainsi qu'entre les conditions A et C ($p < .01$), mais pas entre B et C ($p = .17$). Par ailleurs, une analyse des liens entre variables dépendantes grâce à l'utilisation de corrélations de Pearson indique une corrélation positive modérée entre confiance en son jugement et la pertinence de sa réponse (coefficient r de Pearson = 0.25, $p = .003$). Ensuite, l'entretien réalisé avec chaque participant révèle que tous n'ont pas utilisé l'interface de la même manière. Si certains disent trouver les explications des propositions utiles, d'autres expliquent ne pas les avoir consultées. Ces différents comportements face au système se retrouvent en comparant le nombre total de clics pour les missions réalisées avec l'outil d'aide C pour chaque participant. Bien que l'utilisation moyenne

soit de 18.1 clics, les valeurs s'étendent de 4 à 38 clics, avec un écart-type de 12.4 clics.

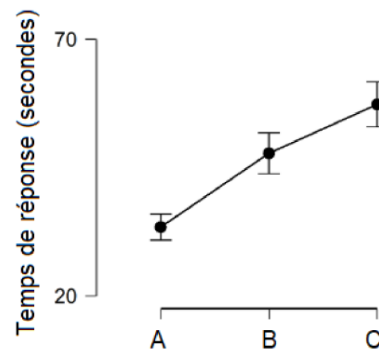


Figure 3 - Temps de jugement des participants en fonction du système d'aide

3 Discussion

Cette étude est basée sur la mise en situation d'opérateurs pour évaluer des propositions de plan faites par un système fortement automatisé. Le micro-monde développé permet à la fois d'avoir des situations contrôlées, où l'optimalité du plan selon des critères prédéfinis peut être effectivement calculée, mais suffisamment complexes pour amener le participant à devoir se faire un avis sur des options difficilement comparables à première vue. Des informations normatives et descriptives étaient proposées pour soutenir le choix du sujet.

Nous avons mis en évidence une influence de la quantité d'explications proposée par le système sur le temps de réponse des participants. Dans la condition A, l'opérateur n'avait que des informations normatives sur la solution considérée comme la plus adaptée par le système. L'ajout d'informations descriptives supplémentaires amène l'opérateur à prendre plus de temps pour évaluer la solution. Plus la quantité d'informations disponible est importante, plus l'opérateur s'y attarde. Il recueille plus d'information sur la situation et les alternatives possibles mais pour autant son jugement sur l'optimalité de la proposition faite par le système ne s'en trouve pas améliorée, ni même la confiance qu'il a dans son jugement. Une des pistes d'explication pourrait être que la question posée (optimalité du choix du trajet) est trop liée à la théorie normative des prises de décision qui ne correspond pas au fonctionnement de l'opérateur, ce qui l'amène à faire confiance au système car il ne se sent pas en mesure d'évaluer ce critère. Une autre explication possible est que l'évaluation et la confiance dans la solution nécessitent une plus grande implication de l'opérateur dans

la construction de la solution, et pas seulement sa visualisation.

Ces résultats nous amènent également à discuter de la pertinence des chemins alternatifs proposés et des informations qui leurs sont associées. Les chemins sous-optimaux avaient été choisis par l'expérimentateur, mais ne correspondent peut-être pas aux alternatives qu'envisage l'opérateur. Dans le cadre applicatif visé, c'est-à-dire de la replanification en présence d'une situation tactique évolutive, une grande incertitude sur les contraintes effectives existe et rend le calcul objectif de la solution optimale impossible. L'ajout d'informations descriptives ne semble pas améliorer le choix de l'opérateur et la confiance qu'il a dans son choix nous amène à envisager que c'est plutôt le mode d'interaction qui peut aider l'utilisateur à mieux appréhender la situation. Ceci nous invite à penser cette phase d'interaction de manière plus dynamique, en permettant par exemple à l'utilisateur de créer des chemins sur la carte pour raffiner son jugement. Des aides correspondant à l'approche constructive présentée en introduction pourraient donc avoir une valeur ajoutée plus importante. Cette piste sera explorée dans une prochaine expérimentation qui permettra de reposer différemment les hypothèses.

On peut aussi noter que le scénario choisi amenait l'opérateur à évaluer une nouvelle planification complète de la mission, alors que dans la réalité on a généralement affaire à une modification d'un plan en cours d'exécution. L'évaluation se fait alors par comparaison avec le plan initial en fonction des modifications des contraintes, ce qui pourrait être plus simple pour l'opérateur. En outre, cette expérimentation a mis en évidence qu'il y avait différentes stratégies d'interaction. Il semblerait que tous les individus ne prennent pas en compte de telles explications de la même façon, certains ne s'y intéressent pas. La phase de débriefing montre aussi que la confiance dans le système est assez forte, les sujets ne détectant pas de trajets grossièrement sous-optimaux. Ceci peut expliquer les comportements qui se reposent sur la décision du système sans même regarder les explications sur la décision prise. Dans la suite de nos travaux, pour mettre en évidence les modalités d'interactions qui favorisent le jugement critique de l'opérateur sur la solution, nous chercherons en particulier à faire entrer l'opérateur dans un processus de co-construction de celle-ci dans un environnement plus complexe nécessitant de faire des compromis entre les différentes contraintes.

4 Bibliographie

- [1] A. Atyabi, S. Mahmoudzadeh, S. Neftimeziani, *Current advancements on autonomous mission planning and management systems: An AUV and UAV perspective*, Annual Reviews in Control, 2018.
- [2] M.J. Barnes, S.G. Lakhmani, E. Holder, J. Chen. *Issues in Human-Agent Communication*, 2019.
- [3] DARPA, *Explainable Artificial Intelligence (XAI)*, Broad Agency Announcement DARPA-BAA-16-53, Arlington, VA, USA, 2016.
- [4] Defense Innovation Board, *AI Principles: Recommendations on the Ethical Use of Artificial Intelligence by the Department of Defense*, 2019.
- [5] V. Demiaux, *Comment permettre à l'homme de garder la main ? - Les enjeux éthiques des algorithmes et de l'intelligence artificielle*, CNIL, 2017.
- [7] G. Klein, R. Hoffman, S. Mueller, *Naturalistic Psychological Model of Explanatory Reasoning: How people explain things to others and to themselves*, International Conference on Naturalistic Decision Making, 2019.
- [8] T. Miller, P. Howe, L. Sonenberg, *Explainable AI: Beware of Inmates Running the Asylum*, arXiv:1712.00547v2, 2017.
- [9] T. Miller, *Explanation in Artificial Intelligence: Insights from social sciences*, Artificial Intelligence, pp. 1–38, 2019.
- [10] A. Nguyen, J. Yosinski, J. Clune, *Deep neural networks are easily fooled: High confidence predictions for unrecognizable images*, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, pp. 427-436, 2015.
- [11] M. Pégny, I. Ibnouhsein, *Quelle transparence pour les algorithmes d'apprentissage machine ?*, hal-01877760, 2018.
- [12] M. Ribera, A. Lapedriza. *Can we do better explanations? A proposal of User-Centered Explainable AI*, Joint Proceedings of the ACM IUI 2019 Workshops, Los Angeles, USA, March 20, 2019.

[13] B. Roy, D. Bouyssou, *Aide Multicritère à la décision : méthodes et cas*, Economica, Paris, 1993.

[14] Task Force IA, *L'Intelligence Artificielle au service de la défense*, 2019

[15] J. Von Neumann, O. Morgenstern, *Theory of Games and Economic Behaviour*, Princeton, NJ, Princeton University Press, 2e édition, 1947.

Transduction sémantique pour la modélisation de système

A. Lamercerie
Univ Rennes, Inria, Irisa

22 juin 2020

Résumé

La conception d'un système est un défi pour les ingénieurs. Le coût d'une erreur s'avère parfois considérable. De fait, leurs détections est un enjeu important. Des méthodes de vérification formelles ont été élaborées pour y répondre. Cependant, l'écart est important entre la promesse théorique de ces méthodes et leur intégration pratique, un écart que cet article vise à réduire. La formalisation automatique d'énoncés exprimés en langue naturelle est explorée par la proposition d'une méthode associant des approches statistiques et symboliques.

Mots-clés

Conception de système, analyse sémantique, représentation abstraite, formalisation.

Abstract

The system design is a challenge for engineers. The cost of an error is sometimes considerable, and their detection is an important issue. Formal verification methods have been developed to address this. However, there is a significant gap between the theoretical promise of these methods and their practical integration, a gap that this article intends to reduce. The automatic formalization of statements expressed in natural language is explored. Our proposal is a method using abstract representation and associating statistical and symbolic approaches.

Keywords

System Design, semantic parsing, abstract representation, formalization.

1 Introduction

La conception d'un système technique commence par un travail d'analyse des besoins et du contexte applicatif. Cette phase se traduit par la rédaction d'un cahier des charges, et nécessite de décrire toutes les caractéristiques fonctionnelles et comportementales du système. Ce document doit être précis, cohérent et complet. C'est une suite d'explications et de règles qui décrivent précisément toutes les propriétés du système à réaliser.

Les concepteurs peuvent être assistés dans leur tâche par des méthodes de spécification et de vérifications formelles, rigoureusement fondées et susceptibles de mettre en évidence des erreurs de conception au coût très élevé si non détectées. Elles offrent potentiellement un avantage indéniable. Pourtant, elles n'ont pas été largement adoptées.

Cela est dû en partie à un certain nombre d'obstacles pragmatiques, tels que l'exigence d'une expertise appropriée, le besoin d'une formation spécifique ou l'absence de support adapté.

Il y a un écart important entre le bénéfice de ces méthodes en théorie, et leur intégration en pratique. Nos travaux visent à réduire cet écart.

Pour atteindre ce but, nous proposons un processus partant des documents produits par le concepteur, et rédigés dans une langue naturelle tel l'anglais. Nous souhaitons extraire de ces documents toutes les informations nécessaires à la construction de modèle formel. Partant d'énoncés informels, l'objectif est donc de produire un énoncé formel équivalent, et intégrable dans une démarche entièrement automatisée.

Le tableau 1 présente quelques exigences d'un cas d'étude spécifiant le fonctionnement d'un système d'accès, avec barrière de sécurité, pour un parking de voitures. Ce cas d'étude, étudié en détail du point de vue "conception de système" dans [3], a été implémenté avec l'outil Mica [6], une bibliothèque qui permet de définir des systèmes complexes en utilisant une syntaxe du langage OCaml. Dans cette mise en oeuvre, la formalisation des exigences a été réalisée manuellement (les exigences ont été traduites dans la syntaxe exigée par l'outil Mica). Notre objectif est d'enrichir ce genre d'outil pour pouvoir traiter directement les exigences exprimées en langue naturelle. Il s'agit de définir une brique qui pourrait y être reliée, un outil formalisant les énoncés dans un format exploitable pour automatiser la génération des modèles.

| <i>Id.</i> | <i>Exigence</i> |
|------------|---|
| E_1 | When a ticket had been issued, the entry gate must open. |
| E_2 | After an exit ticket is inserted, the exit gate must open. |
| E_3 | If the parking is full or a vehicle is passing, the entry gate should not open. |

TABLE 1 – Quelques exigences d'un système d'accès pour un parking de voitures

Nous nous intéressons en particulier aux règles comportementales décrivant les évolutions possibles ou interdites d'un système donné. Pour les besoins de nos expérimentations, nous avons défini un format précis de spécification sous la forme d'un triplet $\langle M, P, C \rangle$. Ces spécifications

traduisent des règles associant un contexte donné C , une modalité M et une propriété P . Le contexte C est une formule logique portant sur un ensemble de propriétés du système, et permettant de cibler les états du système où la règle doit s'appliquer. Si ce contexte est vérifié, la modalité M est attribuée à la propriété P , définissant ainsi la manifestation possible, nécessaire ou interdite d'un événement ou d'une propriété d'état. En associant une sémantique précise à cette représentation, et en combinant un ensemble de règles ainsi formalisées, il est envisageable de modéliser le comportement d'un système à différents niveaux d'abstraction.

Suivant l'état de l'art actuel, les techniques de traitement de langues naturelles ne peuvent garantir une traduction sans erreur, d'autant plus si l'on prend en compte le côté intrinsèquement ambiguë des langues naturelles. Nous pouvons cependant atténuer ce constat avec plusieurs observations sur les documents ciblés. En effet, ces documents sont généralement rédigés avec des formules répétitives, un style restreint et des formules qui se veulent explicites. La mise en évidence d'ambiguïtés serait donc déjà une aide bienvenue, et celles-ci apparaîtraient naturellement avec la construction de modèle incohérent. Les exigences du tableau E_1 et E_3 contiennent une certaine ambiguïté; elles sont en fait inconsistantes, et c'est justement ce type de contradictions qui sont visés.

Du point de vue traitement de la langue, les exigences du tableau 1 illustrent différents phénomènes linguistiques. La condition peut s'exprimer avec des mots clés variés (when, after, if) et une syntaxe différenciée. Ces énoncés traitent de propriétés, et la formalisation de ces énoncés implique d'extraire ces propriétés. Les spécifications contiennent aussi des modalités déontiques, formalisant l'obligation, l'interdiction, la permission et le facultatif. La formalisation d'un énoncé implique la prise en charge contrôlée de tous ces aspects. Plus précisément, il est nécessaire de différencier les entités (gate, ticket), les propriétés (open-gate, insert-ticket) et les groupes nominaux exprimant une relation entre ces éléments ou un phénomène linguistique.

Notre contribution associe deux niveaux d'analyse. Le premier niveau vise la construction de représentation abstraite générique à l'aide d'analyseur syntaxico-sémantique s'appuyant sur des méthodes statistiques. Ces représentations sont ensuite transformées en suivant des opérations de transduction, s'appuyant sur des règles symboliques, jusqu'à obtenir la définition formelle attendue.

La méthodologie proposée applique plusieurs étapes de transductions. Chacune de ces étapes préserve le sens de l'énoncé pour aboutir finalement à la représentation formelle attendue. Plusieurs opérations classiques sont généralement appliquées pour traiter des textes en langue naturelle, telles que le découpage en unités syntaxiques ou l'étiquetage grammatical. Nous appliquons ces opérations en pré-traitement d'une première phase d'analyse syntaxico-sémantique. Celle-ci permet de construire une structure abstraite capturant le sens de l'énoncé. Les représentations de sens abstrait (Abstract Meaning Representation, AMR), introduites par *Banarescu et al.* en 2013 [2], sont des struc-

tures intermédiaires adaptées pour cette étape. Nous exploitons en particulier deux algorithmes. Le premier met en oeuvre un processus itératif à partir d'analyse en dépendance (CAMR, [28]). Le second est fondé sur un modèle prédictif reliant des séquences à des graphes (STOG, [29]). La deuxième phase du traitement consiste à transformer la représentation intermédiaire obtenue pour construire la représentation attendue. La structure abstraite de l'énoncé est analysée selon plusieurs angles, d'abord en s'appuyant sur un référentiel ontologique, puis suivant l'évaluation de requêtes logiques.

La méthodologie proposée a été implémentée et expérimentée sur deux cas d'études, le système d'accès pour un parking et le classique distributeur automatique. Ces deux cas d'études constituent deux corpus totalisant 50 phrases, et couvrant plusieurs phénomènes linguistiques (les conditions, les modalités, les connecteurs temporels et logiques et la négation). Le tableau 2 présente les représentations obtenues sur les exigences du tableau 1. Le taux de correction est de 75 % en moyenne sur les deux corpus expérimentaux.

| <i>Id.</i> | <i>Exigence</i> |
|------------|---|
| E_1 | <necessary, open-gate, issue-ticket> |
| E_2 | <necessary, open-gate, after insert-ticket> |
| E_3 | <possible, not open-gate, full-parking or pass-vehicle> |

TABLE 2 – Formalisation des exigences du tableau 1

2 Travaux connexes

Les méthodes formelles pour la conception de systèmes sont nombreuses. Nous pouvons citer les logiques temporelles, LTL et CTL [8, 7]. Les automates d'interfaces [9] et les interfaces modales [24, 5] visent à fournir une spécification fusionnée du comportement attendu et des environnements possibles. Une variante propositionnelle, les automates d'acceptation propositionnels [17], a été avancée récemment, améliorant l'expressivité pratique de ces méthodes. Les théories de contrats [3], avec une prise en charge native dans plusieurs langages de programmation [21], offrent un cadre flexible en tant que théorie générique.

L'intégration de ces systèmes se heurte à des difficultés pragmatiques que nous voulons dépasser. Notre contribution est un processus automatique assurant le passage de spécifications exprimées en langage naturel vers des spécifications formelles. La vocation de cette proposition est de fournir une interface entre des techniques formelles telles que présentées ci-dessus, et l'énoncé naturel de règles comportementales dans les documents d'ingénierie.

L'usage d'une représentation intermédiaire a été exploré dans plusieurs travaux, avec des approches variées. Une modélisation des connaissances par une ontologie peut être exploitée comme modèle de représentation pivot [26]. L'usage de patrons de spécifications de propriétés temporelles est une approche où le concepteur est guidé par

une grammaire structurée [11, 16]. Les langages contrôlés s’inscrivent dans la même logique. Le langage Attempto Controlled English (ACE, [13]) contraint la structure des phrases tout en restant très proche de l’anglais. Ce langage peut, par exemple, être utilisé pour traduire automatiquement des textes dans la logique du premier ordre.

Le projet ARSENAL [14] fournit un cadre méthodologique de spécifications logiques analysables (LTL). Le système Grammatical Framework [25] est une autre approche combinant langage source et langage cible. Basé sur les grammaires catégorielles et la théorie des types, ce système a été utilisé dans différents projets, et constitue une option intéressante si l’on se limite à un langage naturel contrôlé. Nous posons la problématique un peu différemment. Là où les méthodes ci-dessus offrent un cadre pour construire une interface en langue naturelle pour différentes méthodes formelles, notre démarche vise l’analyse des documents pour en extraire les données nécessaires à l’utilisation entièrement automatisée d’une méthode formelle. Dans ce but, nous nous appuyons sur une représentation pivot. Cette idée est également présente, par exemple, dans des travaux visant la génération de requêtes SQL ou Sparql à partir de langue naturelle [23].

Les grammaires catégorielles combinatoires (CCG, [27]) sont intéressantes dans notre cas pour ses liens avec le lambda-calcul, qui pourrait être exploité comme représentation formelle intermédiaire. Les Grammaires Catégorielles Abstraites [10] en sont une généralisation traitant directement les catégories et le lambda-calcul. La théorie de représentation du discours (DRT, [15]), représentation d’un texte considérant le contenu sémantique qui dépend du contexte, peut être exploitée conjointement aux CCG pour augmenter la couverture des phénomènes linguistiques. Plusieurs analyseurs ont été proposés pour ces formalismes, tel que Ccg2Lambda [4]. Un système de résolution de problèmes mathématiques, acceptant la saisie en langage naturel, a ainsi été conçu en se basant sur les CCG [20].

Notre contribution vise la transformation d’un énoncé, en passant par une représentation sémantique intermédiaire, pour aboutir à une définition plus précise, manipulable et adaptable. Il s’agit d’un processus qui vise à compléter les approches mentionnées. Nous avons fait le choix d’utiliser les représentations de sens abstrait (Abstract Meaning Representation, AMR), un formalisme basé sur les graphes et utilisé pour capturer la sémantique au niveau des phrases [2]. Notre approche pourrait aussi bien être associée au CCG, à l’image des travaux de *Matsuzaki et al* [20]. Elle en est une alternative, dans un contexte différent (la conception de systèmes), mettant l’accent sur les modalités.

Un premier analyseur AMR, introduit par *Flanigan et al* en 2014 [12], produit une analyse syntaxique en se basant sur l’alignement entre mots et concepts. L’identification des concepts est d’abord traitée comme une tâche d’étiquetage de séquence. Les relations sont ensuite identifiées en s’appuyant sur des graphes d’analyse de dépendance. Une approche similaire a été utilisée par *Zhou et al* [30],

où l’identification des concepts et des relations est réalisée conjointement, et *Liu et al* [18], où les alignements sont utilisés comme des variables latentes d’un modèle probabiliste conjoint, avec une amélioration substantielle des résultats obtenus.

L’algorithme CAMR [28] transforme progressivement des analyses de dépendance en AMR, suivant un processus itératif sur les noeuds des arbres de dépendance. D’autres travaux s’appuient également sur cette approche, avec des résultats comparables.

Une méthode inductive, basée sur les grammaires CCG, a également été proposée en exploitant des ressources sémantiques pour convertir des formules logiques en AMR [1]. Plus récemment, un algorithme de prédiction a obtenu de très bons résultats avec un modèle basé attention, reliant séquences et graphes [29].

L’utilisation de ces analyseurs permet de construire des représentations formelles qu’il s’agit maintenant d’exploiter pour obtenir des définitions formelles précises, exploitables avec différentes méthodes de modélisation. C’est ce que nous allons voir dans la prochaine section.

3 Analyse sémantique

Structure intermédiaire entre l’expression naturelle d’un énoncé et leurs traitements par des méthodes automatiques, les représentations sémantiques permettent de capturer formellement la signification d’un texte. L’analyse sémantique proposée ici exploite ces structures. Elle est constituée de plusieurs grandes phases (figure 1). L’idée générale est de partir d’une représentation sémantique intermédiaire, obtenue après une analyse syntaxico-sémantique de l’énoncé initial, puis de transformer cette représentation pour aboutir à la définition formelle attendue.

Premièrement, les données à traiter sont regroupées dans un unique fichier. Les opérations de pré-traitements sont classiques. Elles comprennent le découpage des éléments de chaque phrase en unité syntaxique (token) et l’étiquetage grammatical (POS tagging). Toutes ces opérations sont réalisées avec l’outil Stanford CoreNLP [19].

Dans un second temps, une étape importante du processus consiste en une analyse précise de l’énoncé pour construire une structure formelle qui en capture le sens. Plusieurs travaux récents étudient la construction de telles représentations sémantiques, comme développés dans la section précédente. Nous avons fait le choix d’utiliser les représentations de sens abstrait (Abstract Meaning Representation, AMR), un formalisme basé sur les graphes et utilisé pour capturer la sémantique au niveau des phrases [2].

La construction de cette structure formelle intermédiaire n’est pas une fin en soi, mais un moyen pour aboutir à la définition formelle attendue. L’autre étape importante du processus constitue notre contribution majeure. Elle vise la transformation de la structure intermédiaire dans une démarche contrôlée.

La dernière phase du traitement permet de finaliser le travail en appliquant différentes opérations de finitions sur l’énoncé obtenu. Cette étape n’est pas fondamentale et dé-

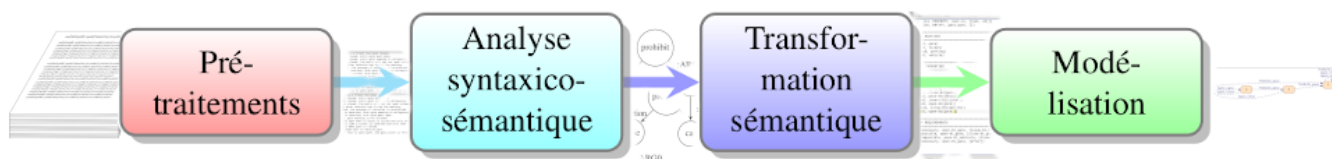


FIGURE 1 – Chaîne de traitements

pend fortement du format de l'énoncé à produire, et de son usage.

Nous avons ainsi défini une chaîne de traitement complète, modulaire et intégrable avec différentes méthodes formelles. Cette brique n'a pas d'intérêt en elle-même. Bien intégrée, en revanche, elle est susceptible de rendre exploitable, en pratique, de nombreux outils ou représentations formelles.

Dans la suite de cette section, nous détaillons les deux phases principales de la chaîne de traitement proposée, la phase d'analyse syntaxico-sémantique, mise en oeuvre pour l'AMR dans nos expérimentations, et la phase de transduction sémantique.

3.1 Analyse syntaxico-sémantique pour produire une représentation intermédiaire

Les représentations de sens abstrait (Abstract Meaning Representation, AMR) permettent de capturer la sémantique au niveau des phrases. Ce formalisme prend la forme d'un graphe où les noeuds correspondent aux concepts présents dans la phrase. Les arêtes orientées reliant les noeuds du graphe traduisent des relations entre ces concepts. Cette structure fournit ainsi une représentation formelle exploitable préservant le sens de l'énoncé.

La figure 2 représente l'AMR de l'énoncé E_1 (voir tableau 1).

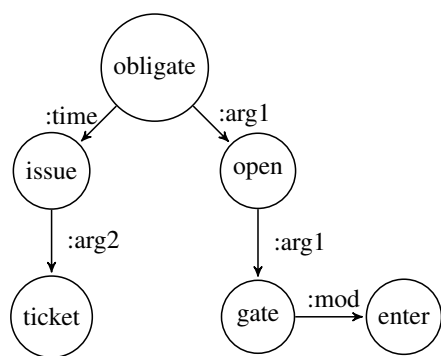


FIGURE 2 – Abstract Meaning Representation (AMR) pour l'énoncé E_1 .

L'analyse syntaxico-sémantique pour AMR est une tâche qui consiste à transformer un énoncé textuel, écrit dans un langage naturel, en une représentation de sens abstrait. Notre outil intègre deux algorithmes [28, 29]. Le premier consiste à transformer progressivement des analyses

de dépendance en AMR, avec un processus itératif sur les noeuds des arbres de dépendance. Le second est plus récent et performant. C'est un algorithme de prédiction utilisant un modèle basé attention, reliant séquences et graphes.

A l'issue de cette étape, l'énoncé a été transformé en une représentation formelle qu'il nous faut maintenant exploiter pour aboutir à la représentation voulue.

3.2 Transduction sémantique

Le processus de transduction sémantique consiste à passer d'une représentation formelle intermédiaire à la représentation cible. La première représentation est obtenue lors de l'étape précédente à partir d'une analyse syntaxico-sémantique de l'énoncé initial. Il s'agit maintenant de transformer la structure obtenue.

L'algorithme proposé comprend plusieurs étapes :

- analyse de la structure abstraite (AMR) pour produire un arbre de syntaxe abstraite (AST)
- analyse de l'AST pour extraire des "données abstraites", en s'appuyant sur une table des symboles et un classificateur
- analyse des données abstraites pour définir une sémantique abstraite sous la forme de trois ensembles (concepts, instances et valuation)
- analyse de la définition abstraite sur une liste de motifs de transduction, associant une requête logique et un modèle pour la représentation à produire

Analyse syntaxique de la structure intermédiaire. La structure abstraite intermédiaire, obtenue en sortie de la phase précédente, constitue la source entrante de la phase de transduction. Nous proposons d'utiliser les techniques classiques mises en oeuvre dans les compilateurs pour analyser cette représentation. En particulier, une définition de syntaxe abstraite (ASD) est proposée pour l'AMR. Celle-ci permet de produire l'arbre de syntaxe abstraite (AST) correspondant à l'énoncé en entrée, et toute une série d'éléments de calcul (table des symboles et données abstraites). La figure 3 montre pour exemple la table des symboles de l'énoncé E_1 . Chaque symbole est associé à une catégorie évaluée à l'aide du classificateur.

Classificateur. Les instances et les relations entre ces instances sont analysées avec un module nommé classificateur. Ce module permet d'attribuer une catégorie à chaque instance, et ainsi de différencier les instances qui couvrent un phénomène linguistique de celles qui définissent une entité (instance d'un concept élémentaire) ou une propriété

```

<corpus_parking_E1> Symbol Table :
(vv1, obligate -01,
 [(ARG2, vv2), (time, vv5)],
 MODALITY)
(vv3, gate, [(mod, vv4)], ENTITY)
(vv2, open -01, [(ARG1, vv3)], PROPERTY)
(vv5, issue -01, [(ARG1, vv6)],
 PROPERTY)
(vv4, enter -01, [], PROPERTY)
(vv6, ticket, [], ENTITY)

```

FIGURE 3 – Table des symboles de l'énoncé E_1 .

(association de concepts). Il s'appuie sur une ontologie regroupant les concepts dans des ensembles structurés, et basé sur un corpus de référence. Nous utilisons la PropBank [22], un corpus annoté de propositions verbales et de leurs arguments. Le choix de la PropBank comme référentiel est directement lié au choix de la représentation intermédiaire, l'AMR s'appuyant également sur ce corpus. L'usage d'un autre référentiel est possible, impliquant néanmoins un alignement entre les concepts de la PropBank, utilisés pour l'AMR, et ceux du corpus choisi.

Définition d'une sémantique abstraite. La définition de la sémantique abstraite définit plusieurs ensembles caractérisant complètement la signification de l'énoncé traité. Il s'agit de l'ensemble de concepts, des instances (associées à différents attributs utiles) et des prédicats vérifiés pour l'énoncé. L'ensemble des prédicats vérifiés définit une valuation de l'énoncé susceptible de vérifier une formule logique. Un exemple est donné, figure 4, pour l'énoncé E_1 .

Application des motifs de transduction. Les motifs de transduction associent une requête logique, et un modèle utilisé pour produire l'énoncé sous la forme attendue. Ils constituent des paramètres du système de transduction. Comme illustré dans le tableau 3, les requêtes sont des formules exprimées dans une forme restreinte de la logique de premier ordre, utilisant uniquement l'opérateur de conjonction logique. Elles permettent de capturer les énoncés qui correspondent à la forme recherchée. Les modèles définissent le format des énoncés à produire en prenant en compte les caractéristiques associées aux instances. De fait, il est possible de modifier la forme des représentations en sortie en modifiant les modèles associés aux requêtes. La définition sémantique de l'énoncé est évaluée pour chaque motif. Si la requête est vérifiée, la sortie est générée en appliquant le modèle aux instances validant la requête.

| | |
|----|---|
| Q. | $\lambda x.\lambda y.\lambda z.isModality(x) \wedge isProperty(y) \wedge \dots$ $\wedge arg + (x, y) \wedge time(x, z) \wedge \dots$ |
| T. | $\langle [1].modality, [2].property, [3].property \rangle$ |

TABLE 3 – Exemple d'un motif de transduction associant une requête (Query) et un modèle (Template).

4 Expérimentations

La chaîne de traitement proposée a été complètement implémentée avec les langages python et java. Le module principal, écrit en python, permet de contrôler les différentes opérations. Stanford CoreNLP, un ensemble d'outils de traitement du langage naturel, est utilisé pour le pré-traitement. L'analyse syntaxico-sémantique pour la génération des structures AMR peut être utilisé en choisissant l'un des deux algorithmes proposés. La mise en oeuvre de ces algorithmes est une adaptation des outils CAMR [28] et STOG [29], dont le code est sous licence libre. Le module ATP (Abstract Transduction Process) est la contribution majeure de notre implémentation. Ce module a été développé en java. Il intègre toutes les étapes de transduction sémantique présentées en section 3.2. Le paramétrage de l'outil comprend la définition de l'ontologie (module OntoMap), utilisée par le classificateur, et la définition des motifs de transduction (module TransductionPattern), utilisée lors de l'étape d'analyse sémantique abstraite. En l'état actuel, seul un fragment du corpus PropBank [22] est intégré dans l'ontologie, fragment suffisant pour couvrir quelques phénomènes linguistiques.

L'expérimentation a été menée sur deux cas d'études, le système d'accès pour un parking de voitures, et le classique distributeur automatique. Ces deux cas d'études forment deux corpus, pour un total de 50 phrases. Ils couvrent plusieurs phénomènes linguistiques tels que les conditions, les modalités, la négation ou encore les connecteurs temporels et logiques.

Les résultats obtenus sont rassemblés dans le tableau 4. La dimension (colonne Dim.) des corpus est précisée. Le résultat 1 (colonne Rés. 1) est le taux de correction observé après la première phase du traitement, tandis que le résultat 2 (colonne Rés. 2) correspond au taux de correction final, évalué après la seconde phase.

| Corpus | Dim. | Rés. 1 | Rés. 2 |
|-----------------|------|--------|-------------|
| Parking Gates | 30 | 0.85 | 0.85 |
| Vending Machine | 20 | 0.87 | 0.67 |
| TOTAL (moyenne) | 50 | 0.87 | 0.77 |

TABLE 4 – Résultats obtenus sur les corpus "Parking Gates" et "Vending Machine".

Les exigences du premier corpus décrivent le fonctionnement d'un système d'accès, avec barrière de sécurité, pour un parking de voitures. Ce cas d'étude a été étudié en détail dans [3] du point de vue "*conception de système*". Le corpus constitué couvre les phénomènes linguistiques ciblés suivant des proportions inégales. Les modalités sont présentes dans 90% des phrases. Les règles sont décrites avec une condition (when, if) dans 73% des cas et avec un connecteur temporel (after, before) pour 27% des phrases. Ce corpus contient également des connecteurs logiques (10%), des négations (20%) et des adverbes (10%).

Le second corpus est constitué de 20 phrases décrivant un système de distribution automatique. Il couvre les


```

<corpus_parking_E1> Abstract Semantic :
concepts = [open-01, issue-01, ticket, obligate-01, gate, enter-01]
instances = [vv1, vv3, vv2, vv5, vv4, vv6]
predicates = [isModality(vv1), isProperty(vv5), mod(vv3, vv4), isEntity(vv3),
  isProperty(vv4), arg+(vv1, vv2), isProperty(vv2), arg1(vv2, vv3), time(vv1, vv5),
  arg+(vv5, vv6), isEntity(vv6), arg2(vv1, vv2), arg1(vv5, vv6), arg+(vv2, vv3)]

```

FIGURE 4 – Définition d’une sémantique abstraite pour l’énoncé E_1 . Les instances sont associées à plusieurs caractéristiques non représentées ici. Les prédicats peuvent être évalués par des requêtes logiques.

mêmes phénomènes linguistiques avec des proportions différentes : 20% pour les conditions, 65% pour les modalités, 55% pour les connecteurs temporels, 15% pour les connecteurs logiques et 35% pour les négations. Il contient également 25% de phrases affirmatives simples n’ayant pas de représentation attendue. La présence de ces phrases montre que la méthode proposée permet de trier automatiquement les données en entrée en ne générant une représentation que lorsque celle-ci a du sens.

5 Conclusion

Cet article s’inscrit dans une progression autour d’un axe visant la formalisation efficace d’énoncés exprimés en langue naturelle. Une première présentation en a été fait en atelier, avec quelques uns des principes envisagés et une implémentation sommaire pour soutenir nos intuitions originales.

Une deuxième étape est validée ici, importante, avec une proposition mieux structurée, une implémentation stable et une expérimentation sur deux corpus certes limités, mais néanmoins complets. Nous avons ainsi montré que la méthode proposée fonctionne, en couvrant plusieurs phénomènes linguistiques avec une certaine efficacité, et avec également un potentiel d’extension.

L’expérimentation menée est néanmoins insuffisante. Nous couvrons quelques phénomènes linguistiques, partiellement. Il est possible d’améliorer le niveau de couverture en enrichissant le référentiel ontologique. De même, le paramétrage des motifs de transduction (requêtes et modèles) peut être complété. Il est également possible, théoriquement, d’appliquer cette méthode pour construire d’autres représentations. Par ailleurs, la méthodologie proposée peut s’appuyer sur une autre représentation intermédiaire, même si ce changement de structure demanderait un effort plus important.

La suite de nos travaux devra permettre de montrer la validité de ces différents points, en menant des expérimentations plus complètes. Au delà, la généralisation de notre démarche pourrait inclure l’apprentissage des motifs de transduction, à partir de quelques exemples positifs, et offrir ainsi la promesse d’un outil plus complet.

Références

- [1] Yoav Artzi, Kenton Lee, and Luke Zettlemoyer. Broad-coverage CCG semantic parsing with AMR.

In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1699–1710. Association for Computational Linguistics, Lisbon, Portugal, September 2015.

- [2] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract meaning representation for semantic banking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186. Association for Computational Linguistics, Sofia, Bulgaria, 2013.
- [3] Albert Benveniste, Benoît Caillaud, Dejan Nickovic, Roberto Passerone, Jean-Baptiste Raclet, Philipp Reinkemeier, Alberto Sangiovanni-Vincentelli, Werner Damm, Thomas A. Henzinger, and Kim G. Larsen. *Contracts for System Design*. 2018.
- [4] Johannes Bos. Open-domain semantic parsing with boxer. *Proceedings of the 20th Nordic Conference of Computational Linguistics*, pages 301–304, 2015.
- [5] Ferenc Bujtor, Sascha Fendrich, Gerald Lüttgen, and Walter Vogler. Nondeterministic modal interfaces. In *SOFSEM 2015 : Theory and Practice of Computer Science*, pages 152–163. Springer Berlin Heidelberg, 2015.
- [6] Benoît Caillaud. Mica : A Modal Interface Compositional Analysis Library, October 2011.
- [7] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. volume 8, pages 244–263. ACM, New York, NY, USA, 1986.
- [8] Edmund M Clarke and E Allen Emerson. Synthesis of synchronization skeletons for branching time temporal logic. In *Logic of programs : Workshop*, volume 131, pages 244–263. 1981.
- [9] Luca de Alfaro and Thomas A. Henzinger. Interface automata. *SIGSOFT Softw. Eng. Notes*, 26(5) :109–120, September 2001.
- [10] Philippe de Groote. Towards abstract categorial grammars. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL ’01, pages 252–259. Association for Computational Linguistics, 2001.

- [11] Matthew B. Dwyer, George S. Avrunin, and James C. Corbett. Patterns in property specifications for finite-state verification. In *Proceedings of the 21st International Conference on Software Engineering, ICSE '99*, pages 411–420, New York, NY, USA, 1999. ACM.
- [12] Jeffrey Flanigan, Sam Thomson, Jaime Carbonell, Chris Dyer, and Noah A. Smith. A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1426–1436. Association for Computational Linguistics, Baltimore, Maryland, 2014.
- [13] Norbert E. Fuchs and Rolf Schwitter. Specifying logic programs in controlled natural language. *CoRR*, abs/cmp-lg/9507009, 1995.
- [14] Shalini Ghosh, Daniel Elenius, Wenchao Li, Patrick Lincoln, Natarajan Shankar, and Wilfried Steiner. abs/1403.3142, 2014.
- [15] Hans Kamp. A theory of truth and semantic representation. In J. A. G. Groenendijk, T. M. V. Janssen, and M. B. J. Stokhof, editors, *Formal Methods in the Study of Language*, volume 1, pages 277–322. Mathematisch Centrum, Amsterdam, 1981.
- [16] Sascha Konrad and Betty H. C. Cheng. Real-time specification patterns. In *Proceedings of the 27th International Conference on Software Engineering, ICSE '05*, pages 372–381, New York, NY, USA, 2005. ACM.
- [17] Aurélien Lamercerie. Une algèbre des automates d'acceptation propositionnelle déterministes comme théorie d'interface pour la conception de systèmes cyberphysiques, 2019. Poster.
- [18] Yijia Liu, Wanxiang Che, Bo Zheng, Bing Qin, and Ting Liu. An amr aligner tuned by transition-based parser. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, page 2422–2430. Association for Computational Linguistics, Brussels, Belgium, 2018.
- [19] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics : System Demonstrations*, pages 55–60. Association for Computational Linguistics, Baltimore, Maryland, June 2014.
- [20] Takuya Matsuzaki, Takumi Ito, Hidenao Iwane, Hirokazu Anai, and Noriko H. Arai. Semantic parsing of pre-university math problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 2131–2141. Association for Computational Linguistics, Vancouver, Canada, July 2017.
- [21] Bertrand Meyer. *Touch of Class : Learning to Program Well Using Object Technology and Design by Contract*. Springer, Software Engineering, 2009.
- [22] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank : An annotated corpus of semantic roles. *Computational Linguistics*, 31(1) :71–106, 2005.
- [23] Camille Pradel, Ollivier Haemmerlé, and Nathalie Hernandez. Natural language query interpretation into SPARQL using patterns. In *Fourth International Workshop on Consuming Linked Data - COLD 2013*, pages pp. 1–12, Sydney, Australia, October 2013.
- [24] Jean-Baptiste Raclet, Eric Badouel, Albert Benveniste, Benoît Caillaud, Axel Legay, and Roberto Passerone. A modal interface theory for component-based design. *Fundamenta Informaticae*, 108(1-2) :119–149, 2010.
- [25] Aarne Ranta. *Grammatical Framework : Programming with Multilingual Grammars*. CSLI Publications, Stanford, 2011. ISBN-10 : 1-57586-626-9 (Paper), 1-57586-627-7 (Cloth).
- [26] Driss Sadoun, Catherine Dubois, Yacine Ghamri-Doudane, and Brigitte Grau. From natural language requirements to formal specification using an ontology. In *IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI 2013)*, pages 755–760, Herndon, VA, United States, 2013.
- [27] Mark Steedman. *The syntactic process*. MIT Press, 2000.
- [28] Chuan Wang, Nianwen Xue, and Sameer Pradhan. A transition-based algorithm for AMR parsing. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 366–375. Association for Computational Linguistics, Denver, Colorado, 2015.
- [29] Sheng Zhang, Xutai Ma, Kevin Duh, and Benjamin Van Durme. Amr parsing as sequence-to-graph transduction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 80–94. Association for Computational Linguistics, Florence, Italy, 2019.
- [30] Junsheng Zhou, Feiyu Xu, Hans Uszkoreit, Weiguang Qu, Ran Li, and Yanhui Gu. AMR parsing with an incremental joint model. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 680–689. Association for Computational Linguistics, Austin, Texas, 2016.

Travail du futur: des portails e-collaboratifs intrinsèquement motivants

M. EL FATRI^{1,2}, F. ROUSSEAU¹, P. DREUX²

¹ Université de Reims Champagne-Ardenne, CReSTIC

² Experconnect, Paris

...

mohamed.elfatri@experconnect.com

Résumé

Les plateformes de travail e-collaboratif permettent aux personnes travaillant sur des sujets similaires de partager un espace de travail, d'échanger du contenu et de faire des recherches « intelligentes ». Cela nécessite un investissement personnel important de la part des utilisateurs qui seuls pourraient freiner les usages individualistes. Cet article propose une première réflexion pour solutionner le problème de manque de motivation des utilisateurs dans le travail collaboratif numérique et sur ce que la personnalisation peut apporter au travail du futur soutenu par des portails e-collaboratifs.

Mots-clés

Personnalisation, Psychologie positive, Transfert des connaissances, Travail collaboratif, Experconnect.

Abstract

E-collaborative work platforms allow people working on similar subjects to share a workspace, exchange content and do « intelligent » searches. This requires a significant personal investment on the part of users who alone could curb individualistic uses. This article offers a first reflection to solve the problem of lack of motivation of users in digital collaborative work and on what the personalization can bring to the work of the future supported by e-collaborative portals.

Keywords

Personalization, Positive psychology, Human-computer interaction, Knowledge transfer, Collaborative work, Experconnect.

1 Introduction

Le travail collaboratif propose aux utilisateurs de s'inscrire dans un principe d'amélioration continue, ainsi qu'un engagement dans une création participative des connaissances. Dans cet objectif des plateformes numériques collaborative ont vu le jour, s'inspirant des travaux de Bhattacharya [1], pour proposer des systèmes modulaires permettant plusieurs fonctionnalités notamment : Réseau Social

d'Entreprise (RSE), gestion de projet, cocréation et Gestion électronique de document (GED) et répondant ainsi à la plupart des besoins fonctionnels des collaborateurs.

Malheureusement, ces environnements ne sont pas très appréciés des utilisateurs et sont confrontés à la non-contribution des individus ce qui représente une limitation handicapante au partage des connaissances (Nous le montrerons par un cas mis en place depuis 2016 dans un cadre professionnel).

Nous pensons que l'association des travaux sur la psychologie positive, en particulier les théories de la motivation à réussir, ainsi que l'intelligence artificielle dans la conception des plateformes de travail numériques du futur facilitera l'innovation et encouragera le partage des connaissances.

2 Problématique

PME parisienne Fondée par Gilles Effront et Caroline Young, Experconnect intervient dans le domaine du management et du conseil en ressources humaines depuis 2005, en France et à l'international. Elle propose à ses entreprises clientes des dispositifs leur permettant de rester bénéficiaires des compétences stratégiques de leurs futurs retraités, qui pourront revenir aux opérations le temps d'une mission d'ingénierie, d'une ouverture de filiale, de la conduite d'un projet ou de l'animation d'une formation.

En 2016, Experconnect a mis en place une extension digitale de son offre « collaboration post-retraite » appelé « KEEN » pour « Knowledge and Expertise Exchange Network » avec un véritable collectif professionnel constitué à la fois de salariés et de retraités. Des experts « partis à la retraite » pouvant s'approprier des cas soumis par des salariés « en poste » et organiser leurs connaissances expertes pour travailler en collaboration à solutionner ces cas.

KEEN permet à l'instar des plateformes de travail collaboratif traditionnelles de répondre aux besoins des utilisateurs en mettant à leurs disposition un espace de discussion dédié aux différentes communautés, des fils d'actualités, des annuaires, un module de chat instantané, un calendrier, une gestion documentaire et différents outils qui sont censés favoriser le partage des connaissances, et stimuler l'intelli-

gence collective.

KEEN est basée sur « JPlatform » de la société Jalios [2], il s'agit d'une solution web qui est facilement accessible partout, et qui repose sur une architecture modulaire. Le cœur de la solution fournit un socle sur lequel sont construits des modules qui viennent enrichir ou modifier fonctionnellement la solution.

Cependant, l'insertion de KEEN dans la pratique s'est faite difficilement. Le taux de participation des effectifs aux sujets de discussion et aux problématiques remontées est très faible en termes de contenu ou des documents postés, voir nul pour certains collaborateurs.

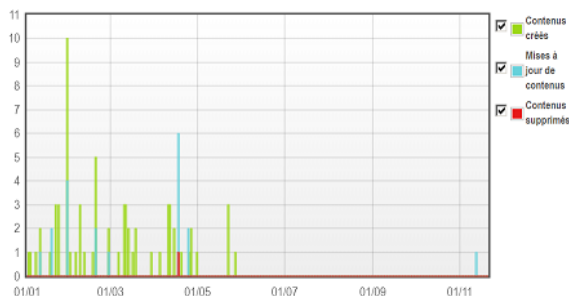


FIGURE 1 – Nombre de contribution sur des contenus en 2019

3 hypothèse

Notre hypothèse place le problème au niveau des utilisateurs : les plateformes e-collaboratives actuelles ne sont pas en mesure de les motiver correctement. Beaucoup de ces systèmes produisent une réaction émotionnelle particulièrement négative chez leurs utilisateurs [3]. Cependant, les émotions détectées sont souvent liées à un manque de motivation [4] qui a pour effet de susciter la peur et la résistance au changement chez les utilisateurs qui manquent de culture et de partage de pratiques plus habitués au développement individuel que collectif. Selon Clifton [5], « la psychologie positive est une révolution de la pensée car elle nous fait passer d'un mode d'expression en termes de déficit vers un mode d'expression en termes positifs ». Cela nous amène à associer travaux de psychologie cognitive et informatique au sens proposé par Hollnagel [6] ou Woods [7] pour concevoir des futurs modèles des plateformes collaboratives digitales. Nous détaillerons par la suite les caractéristiques de certaines théories de la motivation à réussir que nous trouvons intéressantes à prendre en compte lors de la modélisation d'une Plateforme e-collaborative pour différents types de profils utilisateurs.

4 La motivation à réussir : Etat de l'art

Ryan et al. [8] nous précisent qu'être motivé signifie qu'il y a un engagement pour faire quelque chose : si une personne ne ressent ni impulsion ni inspiration pour agir peut ainsi être qualifiée de non motivée, alors que celle qui est stimu-

lée ou activée dans un but est considérée comme motivée. La motivation à réussir est un sous domaine de la motivation qui peut répondre à nos attentes en matière d'aboutissement des tâches de partage des connaissances soutenues par des espaces numériques de travail. Plusieurs théories qui ont vu le jour -qu'on listera par la suite- ont tenté de répondre aux trois questions principales : • Est-ce que je peux faire cette tâche ? • Est-ce que je veux faire cette tâche ? • Que dois-je faire pour réussir dans cette tâche ?

4.1 Théorie de l'auto-efficacité (Self Efficacy Theory)

Elaborée par Bandura [9], dans le cadre théorique plus large de sa théorie sociale cognitive [10], elle met l'accent sur le sentiment qu'une personne peut produire sachant qu'elle peut faire les choses bien et efficacement dans n'importe quel contexte. En spécifiant deux types d'espérances, croire que certains comportements, et/ou qu'on est capable de réaliser les comportements nécessaires pour produire le résultat attendu, Bandura estime que l'efficacité personnelle puise à quatre sources : les performances antérieures, le plaisir retiré de l'apprentissage, les encouragements verbaux des autres personnes, et les réactions physiologiques et émotionnelles. La théorie de l'auto-efficacité a été appliquée et validée dans des nombreux domaines, essentiellement l'apprentissage humain. De ce fait si nous considérons un environnement de travail collaboratif numérique est aussi un environnement d'apprentissage humain et de partage de connaissances, en intégrant des encouragements verbaux et sans omettre le plaisir retiré de la participation à ce partage, ne fera qu'améliorer la création participative des connaissances et l'innovation collaborative.

4.2 Théorie de l'auto-détermination

La Théorie de l'Autodétermination (TAD) [11] stipule que plusieurs différents types de motivation existent (Motivation intrinsèque, Motivation extrinsèque, et Amotivation) qui peuvent provoquer un comportement plus ou moins efficace, également il existe trois besoins de compétence, de relationnel et d'autonomie, ces besoins partagés universellement indépendamment des cultures [12], [13], [14]. Conformément à Ryan [15], plus le type de la motivation pour une tâche tend vers la motivation intrinsèque, meilleure est la qualité de la motivation et par conséquence l'engagement dans l'activité. La théorie de l'auto-détermination a été appliquée et validée dans des nombreux domaines, dont la santé, le sport, et surtout l'apprentissage humain.

4.3 La théorie du Flow

Csikszent Mihalyi [16] dans le cadre de ses études sur la créativité et le bonheur a désigné sous le terme « Flow » une expérience consciente, positive et complexe qui exprimait le sentiment de fluidité et de continuité ainsi que l'importante concentration que décrivaient les personnes interrogées lorsqu'elles exerçaient leur activité préférée. Parmi les circonstances qui sont supposées conduire à l'expérience

optimale, on trouve l'équilibre compétences/défi, la clarté des objectifs et un retour instantané. Le « Flow » se caractérise par un but clair, une rétroaction immédiate, l'intégration de la conscience et de l'action, la concentration sur la tâche, la perception de contrôle de la situation, l'absence de préoccupation à propos de soi, un sens altéré du temps, la motivation intrinsèque et – condition fondamentale – l'équilibre entre les exigences de la tâche et les capacités de l'individu [17], [18], et [19].

Cet état de l'art nous a permis de mettre en évidence les facteurs menant à un état psychologique positive et que doivent être pris en compte dans la personnalisation des plateformes e-collaboratifs :

TABLE 1 – Besoins pour la motivation à réussir.

| | Auto- efficacité | Auto- détermination | Flow |
|-------------------------|---------------------|------------------------|------|
| Compétences | X | X | X |
| Plaisir | X | | X |
| Relationnel | X | X | |
| Réactions | X | | |
| Autonomie | | X | |
| Clarté des objectifs | | | X |

5 Conclusion

À ce jour peu d'études ont demandé quels sont les besoins psychologiques à prendre en compte lors de la conception d'un environnement numérique de travail collaboratif. La prise en charge de ces besoins est primordiale pour rendre une plateforme e-collaboratif plus attractive, ce qu'est garantira un partage des connaissances plus large et par conséquence enrichir l'innovation. Ainsi, dans un but motivationnel, la prochaine étape sera de concevoir un modèle de personnalisation et recommandation motivationnelle. Intégré à « KEEN », ce dernier doit être capable de :

1. Prendre en compte les compétences des utilisateurs.
2. offrir du plaisir durant l'utilisation.
3. Avoir une dimension relationnelle humaine entre différents utilisateurs : encouragements...
4. Prendre en compte les réactions psychologique et émotionnelle.
5. offrir une grande autonomie aux utilisateurs.
6. Avoir des objectifs (Tâche) très clairs et précis.
7. s'adapter en fonction des utilisateurs en offrant des Défis en lien avec leurs compétences.

L'ensemble sera testé en collaboration avec un collectif constitué des experts retraités, des salariés en poste des sociétés Framatome et EDF, ainsi que des collaborateurs de la société Experconnect.

Références

- [1] Bhattacharya Madhumita and Chatterjee Ranajit, Collaborative innovation as a process for cognitive development, *Journal of Interactive Learning Research*, Vol. 11, pp. 295-312, 2000.
- [2] Jaliou, Jaliou JPlatform 10.0 Le socle digital de l'organisation *Livre Blanc Fonctionnel*, 2018.
- [3] O'regan and Kerry, Emotion and e-learning, *Journal of Asynchronous learning networks*, Vol 7, pp. 78-92, 2003.
- [4] Weiner and Bernard, An attributional theory of achievement motivation and emotion, *Psychological review*, Vol. 92, pp. 548, 1985.
- [5] Clifton D. (2002), Positive Psychological Assessment, *A handbook of Models and Measures. American Psychological Association*
- [6] Hollnagel, Erik and Woods, David D, Cognitive systems engineering : New wine in new bottles, *International Journal of Man-Machine Studies*, Vol 6, pp. 583-600, 1983.
- [7] Woods, David D and Roth, Emilie M, Cognitive systems engineering, *Handbook of human-computer interaction*, pp 3-43, 1988
- [8] Ryan, Richard M and Deci, Edward L, Intrinsic and extrinsic motivations : Classic definitions and new directions, *Contemporary educational psychology*, Vol 25, pp. 54-67, 2000.
- [9] Bandura and A, Self-efficacy. I : VS Ramachandran (red.), *Encyclopedia of human behavior*, Vol. 4, pp. 71-81,1994.
- [10] Bandura and Albert, Social foundations of thought and action, *Englewood Cliffs, NJ*, Vol 1986,1986
- [11] Ryan, Richard M and Deci, Edward L, Self determination theory and the facilitation of intrinsic motivation, social development, and well-being, *American psychologist*, Vol. 55, pp. 68,2000.
- [12] Chirkov, Valery I and Ryan, Richard M, Parent and teacher autonomy-support in Russian and US adolescents : Common effects on well-being and academic motivation, *Journal of cross-cultural psychology*, Vol. 32, pp. 618-635, 2001.
- [13] Deci, Edward L and Ryan, Richard M, Overview of self-determination theory : An organismic dialectical perspective, *Handbook of self-determination research*, pp. 3-33, 2002
- [14] Chirkov, Valery and Ryan, Richard M and Kim, Youngmee and Kaplan, Ulas, Differentiating autonomy from individualism and independence : A self-determination theory perspective on internalization of cultural orientations and well-being, *Journal of personality and social psychology*, Vol. 84, pp. 97, 2003.
- [15] Ryan, Richard M and Deci, Edward L, On happiness and human potentials : A review of research on hedonic and eudaimonic well-being, *Annual review of psychology*, Vol. 52, pp. 141-166,2001.

- [16] Csikszentmihalyi, Mihaly, Flow and education, *NAMTA journal*, Vol. 22, pp. 2-35, 1997.
- [17] Fullagar, Clive J and Knight, Patrick A and Sovern, Heather S, Challenge/skill balance, flow, and performance anxiety, *Applied Psychology*, Vol. 62, pp. 236-259, 2013.
- [18] Kawabata, Masato and Mallett, Clifford J, Flow experience in physical activity : Examination of the internal structure of flow from a process-related perspective, *Motivation and Emotion*, Vol. 35, pp. 393-402, 2011.
- [19] Moneta, Giovanni B and Csikszentmihalyi, Mihaly, The effect of perceived challenges and skills on the quality of subjective experience, *Journal of personality*, Vol. 64, pp. 275-310, 1996.

Increasing interoperability in the Web of Things using Autonomous Agents (Position Paper)

Edison Chung¹

Maxime Lefrançois¹

Olivier Boissier¹

¹ Mines Saint-Etienne, Univ Lyon, Univ Jean Monnet, IOGS, CNRS, UMR 5516, LHC,
Institut Henri Fayol, F - 42023 Saint-Etienne France

{edison.chung, maxime.lefrancois, olivier.boissier}@emse.fr

Résumé

Le Web des objets (WoT) vise à permettre l'interopérabilité au sein de l'Internet des objets (IoT) et des domaines d'application. Toutefois, le nombre et l'hétérogénéité des dispositifs connectés ne cessent de croître. L'intégration de multiples objets avec des méthodes différentes d'interaction nécessite des efforts et des interventions humaines, et est de plus en plus difficile avec leur hétérogénéité croissante. Une certaine forme d'automatisation de cette tâche, par exemple en déléguant de nouvelles compositions de services à un agent autonome, nécessite des mécanismes d'accès et d'interaction uniformes entre les Objets. Dans cet article, nous étudions comment les solutions d'interopérabilité sémantique (SIS) peuvent être combinées avec des systèmes multi-agents (SMA), pour permettre aux agents d'interagir de manière autonome avec ces ressources du WoT. Nous utilisons comme point de départ les travaux du groupe de travail sur le Web des objets au W3C (World Wide Web Consortium). Nous identifions et décrivons les problèmes liés i) à la représentation, ii) à la découverte et à la sélection des objets, et iii) à l'interaction entre les agents et les objets. Les réponses à ces questions sont essentielles pour assurer cette intégration en utilisant les SMA, le WoT et SIS.

Mots Clef

Web des objets, interopérabilité, systèmes multi-agents, Thing Description, agents autonomes.

Abstract

The Web of Things (WoT) aims to enable interoperability across the Internet of Things (IoT) and application domains. However, the number and heterogeneity of connected devices continues to grow. The integration of multiple Things with different interaction methods requires human effort and intervention, and is increasingly challenging as their heterogeneity grows. Some form of automation of this task, for example by delegating new service compositions to autonomous agent, would require mechanisms for uniform access and interaction among Things. In this posi-

tion paper, we investigate how Semantic Interoperability Solutions (SIS) can be combined with multi-agent systems (MAS), to allow agents to autonomously interact with these WoT resources. We use as a starting point the work of the Web of Things Working Group at W3C (World Wide Web Consortium). We identify and describe issues related to i) the representation, ii) the discovery and selection of Things, and iii) the interaction between agents and Things. We claim that these issues are the core tenets to ensure this integration using MAS, WoT and SIS.

Keywords

Web of Things, Interoperability, Multi-Agent Systems, Thing Description, Autonomous Agents.

1 Introduction

While the number of devices, or Things, connected to the Internet of Things (IoT) grows every day, the IoT suffers from a lack of interoperability across platforms [16]. In order to develop technology-independent networked applications, platform independent APIs are needed, as well as means for different platforms to discover how to interoperate with one another. The Web of Things (WoT) [23] is being developed to address these issues and to be the information space built upon the IoT where new services (physical and digital) are spawned, building an environment with inter-connected objects participating to applications such as smart buildings, intelligent transport, smart energy management, etc. Things in the WoT are physical or virtual entities whose interaction affordances are described. They can be connected devices, but also entities such as people, places, and abstract concepts (e.g. events) [12].

Things are heterogeneous in nature, with different ways to interact with them. Thus, creating and providing new services based on the composition of these Things proves to be a complex task for developers or mashup users. To reduce this complexity, we need to introduce automatic composition and thus rely on mechanisms to ensure interoperability, and provide uniform access to Things.

In this paper, we propose to combine Semantic Interopera-

bility Solutions (SIS) and multi-agent systems (MAS) to bring the required autonomy and interoperability on top of the WoT. Multi-agent technologies introduce separation of concerns between autonomous decision entities (agents) and active or passive entities (artifacts) [2]. This is done through uniform interfaces that agents can use to perceive and act on artifacts. Artifacts can be used to encapsulate Things and act as proxies for agents to access the real world via the WoT.

Agents could autonomously and uniformly interact with Things, in other words, discover, reason, access these Things, and build added value services from their composition. In order to achieve this objective, several challenges have to be considered. This paper aims to state and illustrate these challenges.

In section 2, we briefly describe related concepts, definitions and the state of the art in WoT, SIS, and MAS. In section 3, we describe the problem and the related challenges to address the autonomous interaction among things. Finally, in section 4, we conclude our work and discuss perspectives.

2 Background

The WoT is the information space built upon the IoT. Its aim is to enable interoperability across IoT Platforms and application domains [12]. It does so by providing i) IoT interfaces to allow IoT devices and services to communicate with each other, and ii) standards to define and program IoT behavior. Things, the central building block of the WoT, are abstractions of physical or virtual entities [12] that need to be represented in IoT applications. A Thing may provide a network-facing API for interaction (WoT Interface).

The WoT Thing Description (TD) specifies the semantics of these interactions. This semantics is defined using an ontology : an explicit specification of a conceptualization [8]. TD is a central building block in the W3C Web of Things (WoT) and can be considered as the entry point of a Thing. The core vocabulary for the W3C WoT TD is shown on Figure 1. A TD comprises the description of the Interaction Affordances of the Thing, general metadata such as communication and security mechanisms, and potentially other domain specific metadata [12]. Three categories of Interaction Affordances are defined : Properties, Actions, and Events [11]. Properties can be used for sensing and controlling parameters. Actions model invocation of physical processes, and abstract calls of existing platforms. Events are used for the publish/subscribe model of communication where data are sent asynchronously to the consumer.

A Multi-Agent System (MAS) is a loosely coupled network of agents that work together to solve problems beyond the individual capabilities or knowledge of each agent [6]. The characteristics of a MAS [10], are :

1. locality and limited information and capabilities in each agent,
2. no global view and system control,

3. data distribution and control decentralization,
4. asynchronous computation.

In the context of MAS, we focus on the Multi-Agent Oriented Programming approach [2] that makes a clear separation of concerns between agents and their environment.

An agent is a physical or software entity, situated in its environment. It is capable of flexible autonomous action in order to meet its design objectives. Autonomy for an agent means that it is able to decide on its actions without direct intervention from humans [10]. An agent can sense the environment and perform actions to change it. The environment denotes the conceptual space where agents execute and interact. Artifacts encapsulate any kind of resource or tool (web service, sensor, actuator) that agents can use in the environment to achieve their goals [17]. They provide uniform access through a set of observable properties, operations, and signals. Agents can perceive artifacts properties and signals, and autonomously decide to execute artifacts operations.

While MAS and IoT have both existed for a long time, few integration attempts have been realized. One notable paradigm is the Internet of Agents (IoA) [20]. It uses agents to represent and streamline interactions of Things. The Agent of Things (AoT) [15] and the Multiagent Web [13] are two approaches that adopt this paradigm. In the AoT, each Thing embeds an autonomous agent. The hardware and complexity requirements hamper the reusability and scalability of this approach [15]. The Multiagent Web uses a multi-layer MAS to govern a group of devices. This approach is scalable as it is built on top of the Web architecture [13].

3 Challenges

As we can see from the previous section, even if several bricks exist, there doesn't exist yet a solution to the problem which is considered in this paper : *How can we allow agents to autonomously and uniformly interact with Things in the Web of Things ?*

To better illustrate it, let us consider a scenario in which we want to optimize the power consumption of a smart building, composed of several spaces in which devices (e.g. sensors, thermostats, lights, fans, ...) are deployed. Our goal is to introduce autonomous agents to control the devices and optimize the smart building energy consumption. Our first challenge is to introduce the devices affordances to the agents, since we consider that the existing devices in the building are, at the beginning, external entities to the agents environment. In other words, we need a uniform representation for the devices in this environment. Our second challenge is to let the agents autonomously discover and select the devices that will help them to achieve their goals. Our third challenge is to allow the agents to operate devices, which implies that agents interact with devices through their representation.

This scenario introduces the following functional issues that need to be addressed to integrate MAS with WoT :

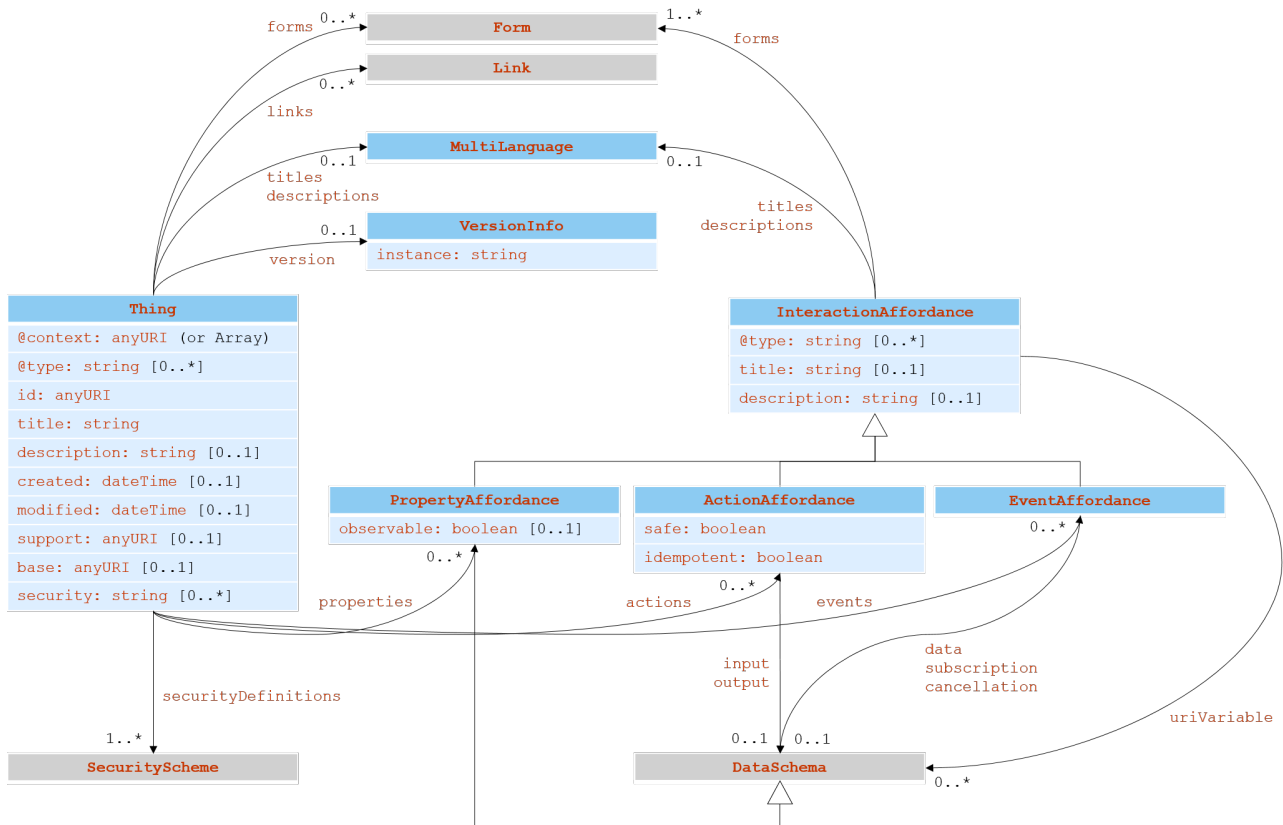


FIGURE 1 – W3C WoT TD core vocabulary

1. representation of Things,
2. discovery and selection of Things, and
3. interaction with Things.

Each issue is separately described below.

3.1 How can we represent Things in the agents' environment?

Each Thing has its own properties, events, and actions; described by its TD, which may potentially include additional metadata based on other ontologies.

Consider we have two Things : the thermostat, and the room its in. The *thermostat Thing* is a connected device. It should thus expose to the agent i) its *current temperature property*, ii) an event *temperature reached* when some temperature is reached and iii) a *change temperature action*. The *room Thing* is not a connected device, so it will only expose the following properties *room name* and *floor number* for instance.

The representation of Things has two objectives :

1. to provide information to the agents about the Thing, and
2. to provide a way for the agents to interact with it; access to its properties and events and using its actions.

The representation should be self-sufficient ; the agent does not require any other representation to fully interact with the Thing. Furthermore, a uniform representation can mask the heterogeneity between Things.

Related to representation, several works have proposed using artifacts to offer agents an access to external resources such as ontologies [7] and algorithms [3]. Simply encapsulating external resources in artifacts is not enough. A standardized way to describe them is necessary so that agents can reason on their usage interface. Semantic descriptions, such as OWL [22] for ontologies and the TD [11] for Things, may prove useful in this regard. The adoption of TD has not been widespread, since it is a relatively new concept.

A good approach would be to model Things as artifacts, so that agents can easily interact with them. This approach requires aligning the concept and description of artifacts in agents environment with the Thing Description. An artifact provides a set of properties, a set of signals and a set of operations [2]. Two Things may not have a similar TD. All TDs share the same general structure. A similar comment applies also to artifacts. Thus, aligning the TD structure to the structure of a generic artifact becomes the most important task to achieve. Ideally, a complete alignment would allow us to represent anything as an artifact by simply giving it a TD. Since artifacts are situated in the agents' en-

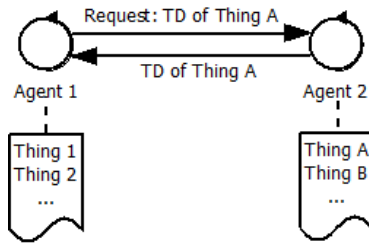


FIGURE 2 – Agents with their own list of Things

vironment, there is no need for modifications to the Thing itself.

3.2 How can agents discover and select Things to achieve their goals?

Asking developers to prepare a list of TDs for the agents will become impractical as the number of Things increases. This list provides a degree of uniformity, but we believe that agent autonomy for discovery of Things is reduced. The agent must be autonomous in its searching and discovering Things. It should be able to choose Things to interact with, according to their own goals without human intervention.

Automated Thing discovery in the WoT has been investigated in several proposals. For instance, Dynamix [3] enables conventional Web apps to discover and control Universal Plug-and-Play (UPnP) and AirPlay devices uniformly. In [21], a middleware discovers Bluetooth and UPnP devices, wraps them with a semantic service description, and shows the available services through a software client. Semantic services, such as Uberdust [18], SPITFIRE [19], and DiscoWoT [14], have as main objective to facilitate the discovery, selection, and utilization of Web-enabled devices. However, applications built on top of these platforms are not interoperable as they do not use a common ontology to describe Things.

Let us notice that in the multi-agent domain, several attempts have been realized. For instance, in [5], artifacts are implemented as part of an e-learning system. They provide search and retrieval capabilities to agents for i) modeling learner information using a learner model ontology, and ii) retrieving resources from learning object repositories.

A possible approach for this challenge would be to allow the agent itself to handle the discovery of Things. The agent can manage its own list of Things. The agent can select the Things from this list which will help achieve its goal. This would also allow for a network of agents to be able to share information about the Things as depicted in Figure 2. This could be achieved, for example, by using peer-to-peer communication enabled by JADE [9].

Another approach would be to create or adapt a repository to store TDs [4], as shown in Figure 3. When introducing a new agent, it only needs to know that there is a repository and that it can use queries to obtain a list of Things. The agent can send a query for specific interaction affor-

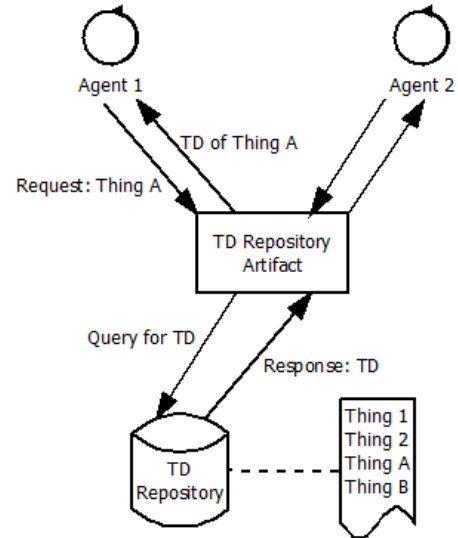


FIGURE 3 – Agents query a TD repository

dances, devices, metadata, etc., in order to select Things. It is important to note that the repository would require its own representation in the agents environment. This could be done through either an agent or an artifact. It must be capable of automated discovery of Things.

3.3 How can agents interact with Things?

Agents should be able to discover and interact with Things uniformly all throughout the WoT. State-of-the-art approaches about IoA ([15], [13]) rely on agents as mediators between Things and other agents. Interactions between agents are more complex than interactions between agents and resources [1].

As mentioned in section 3.1, we can encapsulate different Things using the same generic artifact. This would allow any agent to easily discover and interact with any instantiated Thing based on this artifact. The agent should be able to retrieve properties of a Thing as if they were the artifacts properties. The agent should call the actions and events as if they were the artifacts operations. Consider the example of a room with a thermostat and a heating system, as shown in Figure 4. An agent reads the property corresponding to the room's current temperature. The agent calls the action of the heating system which regulates the temperature of the room. When the desired temperature is reached, the thermostat sends a notification event in order to tell the agent that the action is complete. The agent may obtain information about the room by reading properties such as the room number and corresponding floor. Since relations between Things can be expressed in their TD using Links, the agent should also be able to know that the thermostat is in this specific room.

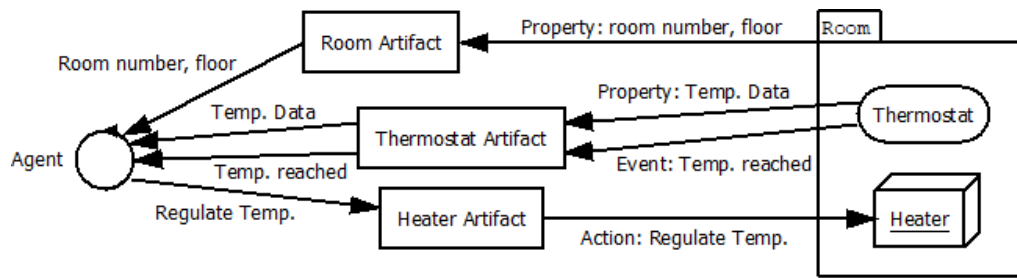


FIGURE 4 – Interaction example between the agent and Things

4 Conclusion and perspectives

In this paper, we provided an overview of challenges that need to be addressed in order to allow autonomous agents to interact with Things in the WoT. We analyzed issues for each of these challenges. We also presented a general overview of the state of the art for the integration between MAS and the WoT, and provided some examples of semantic services for Thing discovery in the WoT.

We discussed the potential use of artifact encapsulation to solve the challenge of representation of Things in the agents environment, as well as the issue of interaction between agents and Things. The practical implementation of this concept will be done in future work. We discussed possible approaches for autonomous discovery and selection of Things by agents. Issues that were not discussed in this paper, such as the management of Things by agents, could be considered for future work.

4.1 Acknowledgement

This work was partially funded by ENGIE Lab CRIGEN. We thank Lynda TEMAL and Sarra BEN-ABBES whose comments/suggestions helped improve and clarify this manuscript.

Références

- [1] D. L. Acay, G. Tidhar, and L. Sonenberg, "Extending agent capabilities : tools vs. agents," in *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 2. IEEE, 2008, pp. 259–265.
- [2] O. Boissier, R. H. Bordini, J. Hübner, and A. Ricci, "Dimensions in programming multi-agent systems," *The Knowledge Engineering Review*, vol. 34, 2019.
- [3] D. Carlson, B. Altakrouri, and A. Schrader, "An Ad-Hoc Smart Gateway Platform for the Web of Things," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, Aug. 2013, pp. 619–625.
- [4] E. Chung, M. Lefrançois, O. Boissier, "Bringing Connected Objects to Autonomous Agents operating on the Web," 2019.
- [5] B. Ciloglugil and M. M. Inceoglu, "An Adaptive E-Learning Environment Architecture Based on Agents and Artifacts Metamodel," in *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, Jul. 2018, pp. 72–74.
- [6] E. H. Durfee and V. R. Lesser, "Distributed Artificial Intelligence (Vol. 2)," M. Huhns, Ed. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1989, pp. 229–243.
- [7] A. Freitas, A. R. Panisson, L. Hilgert, F. Meneguzzi, R. Vieira, and R. H. Bordini, "Integrating ontologies with multi-agent systems through CArtAgO artifacts," in *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 2. IEEE, 2015, pp. 143–150.
- [8] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, Jun. 1993.
- [9] JADE Team, "Java Agent DEvelopment Framework." [Online]. Available : <https://jade.tilab.com/>
- [10] N. R. Jennings, K. Sycara, and M. Wooldridge, "A Roadmap of Agent Research and Development," *Autonomous Agents and Multi-Agent Systems*, vol. 1, no. 1, pp. 7–38, Mar. 1998. [Online]. Available : <https://doi.org/10.1023/A:1010090405266>
- [11] S. Kaebisch, T. Kamiya, M. McCool, V. Charpenay, and M. Kovatsch, "Web of Things (WoT) Thing Description," W3C, W3C Working Draft, Oct. 2018.
- [12] M. Kovatsch, R. Matsukura, M. Lagally, T. Kawaguchi, K. Toumura, and K. Kajimoto, "Web of Things (WoT) Architecture," W3C, W3C Working Draft, May 2019.
- [13] P. Leong and L. Lu, "Multiagent Web for the Internet of Things," in *2014 International Conference on Information Science Applications (ICISA)*, May 2014, pp. 1–4.
- [14] S. Mayer and D. Guinard, "An Extensible Discovery Service for Smart Things," in *Proceedings of the Second International Workshop on Web of Things*, ser. WoT '11. New York, NY, USA : ACM, 2011, pp. 7 :1–7 :6, event-place : San Francisco, California, USA.

- [15] A. M. Mzahm, M. S. Ahmad, and A. Y. C. Tang, “Agents of Things (AoT) : An intelligent operational concept of the Internet of Things (IoT),” in *2013 13th International Conference on Intelligent Systems Design and Applications*, Dec. 2013, pp. 159–164.
- [16] M. Noura, M. Atiquzzaman, and M. Gaedke, “Interoperability in internet of things infrastructure : Classification, challenges, and future work,” 10 2018.
- [17] A. Omicini, A. Ricci, and M. Viroli, “Artifacts in the A&A Meta-model for Multi-agent Systems,” *Autonomous Agents and Multi-Agent Systems*, vol. 17, no. 3, pp. 432–456, Dec. 2008.
- [18] A. Orestis, A. Dimitrios, and C. Ioannis, “Towards integrating IoT devices with the Web,” in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, Sep. 2012, pp. 1–4.
- [19] D. Pfisterer, K. Romer, D. Bimschas, O. Kleine, R. Mietz, C. Truong, H. Hasemann, A. Kröller, M. Pagel, M. Hauswirth, M. Karnstedt, M. Leggieri, A. Passant, and R. Richardson, “SPITFIRE : toward a semantic web of things,” *IEEE Communications Magazine*, vol. 49, no. 11, pp. 40–48, Nov. 2011.
- [20] P. Pico-Valencia, J. Holgado-Terriza, D. Herrera-Sánchez, and J. Sampietro, “Towards the Internet of Agents : An Analysis of the Internet of Things from the Intelligence and Autonomy Perspective,” *Ingeniería e Investigación*, vol. 38, pp. 121–129, 2018.
- [21] Z. Song, A. A. Cardenas, and R. Masuoka, “Semantic middleware for the Internet of Things,” in *2010 Internet of Things (IOT)*, Nov. 2010, pp. 1–8.
- [22] W3C, “OWL 2 web ontology language document overview (second edition),” W3C, Tech. Rep., Dec. 2012, <http://www.w3.org/TR/2012/REC-owl2-overview-20121211/>.
- [23] W3C, “W3C Web of Things at W3C.” [Online]. Available : <https://www.w3.org/WoT/>

Modèle d'Interruption pour la Répartition des Tâches : Application à une simulation de colonie d'abeilles.

Thomas Alves¹, Jérémy Rivière¹, Cédric Alaux², Yves Le Conte²
Frank Singhoff^{1,4}, Thierry Duval³, Vincent Rodin¹

¹ Univ Brest, Lab-STICC, CNRS, UMR 6285, Brest, France
{prenom.nom}@univ-brest.fr

² INRAE, UR 406 Abeilles et Environnement, Avignon, France
cedric.alaux@inrae.fr, yves.le-conte@inrae.fr

³ IMT Atlantique, Lab-STICC, CNRS, UMR 6285, Brest, France
thierry.duval@imt-atlantique.fr

⁴ Groupement de Défense Sanitaire Apicole du Finistère (GDSA29), France

Résumé

Les principaux modèles d'allocation de tâches reposent sur l'hypothèse que chaque tâche est associée à un stimulus, et que l'exécution de cette tâche réduit ce stimulus. Dans cet article, nous proposons un modèle basé sur les modèles classiques "à seuil" (Fixed Threshold Models) et sur la motivation intrinsèque de l'agent, qui permet de traiter des tâches déconnectées de stimulus. Chaque agent utilise cette motivation, influencée par ses performances dans la réalisation de la tâche, pour évaluer la priorité de cette tâche et décider de l'interrompre ou de la poursuivre. Nous avons implémenté ce modèle afin de simuler la répartition des tâches dans une colonie d'abeilles simplifiée. Les premiers résultats montrent que la répartition des tâches est efficace et robuste, et que l'utilisation de la motivation intrinsèque comme mécanisme d'interruption semble l'améliorer.

Mots-clés

Simulation à base d'agents, Répartition des tâches, Auto-Organisation, Insectes sociaux.

Abstract

Task allocation model often rely on the assumption that each task is associated to a stimulus, and executing this task will lower the intensity of that stimulus. We propose here a model based on response thresholds model associated with intrinsic motivation and internal states of the agents able to handle tasks dissociated of stimulus. Each agent will estimate how important is its current task based on its results and decide whether to interrupt or pursue its current task. Preliminary work shows that task allocation is effective and that our interruption model seems to improve the overall results.

1 Introduction

La capacité des insectes sociaux à répartir leur main-d'œuvre sans aucun contrôle central est étudiée depuis plus d'une vingtaine d'années. Cette capacité d'adaptation dy-

namique à leur environnement permet aux insectes sociaux d'être résistants aux changements et de continuer à prospérer. En informatique, et plus particulièrement dans les systèmes multi-agents, les modèles de répartition des tâches adaptés des insectes sociaux sont nombreux et efficaces, utilisés par exemple en simulation [1], [2] et pour la résolution générale de problèmes [3], [4]. Les modèles de répartition des tâches reposent principalement sur deux hypothèses : 1) chaque tâche est associée à un stimulus et 2) l'exécution de cette tâche abaisse le niveau de son stimulus, alors que sans exécution ce stimulus augmente [5]. Un agent percevant ainsi un stimulus externe (dans l'environnement) ou interne (relatif à son état physique) déclenche l'exécution de la tâche associée, ce qui diminue l'intensité du stimulus et la nécessité de refaire cette même tâche. Ainsi, le stimulus perçu peut être considéré comme une représentation du travail nécessaire à la tâche.

Cependant, dans certains cas, l'exécution d'une tâche n'affecte pas l'intensité du stimulus perçu associé, ou il y a même une absence totale de lien entre une tâche et un stimulus. Dans ce cas, nous devons trouver d'autres moyens de piloter l'algorithme d'attribution des tâches. Comment un agent décide-t-il de s'engager dans une tâche ? Comment décide-t-il de continuer à exécuter la tâche en cours ou de l'interrompre ?

Dans cet article, nous décrivons un modèle capable d'attribuer à des agents des tâches sans lien direct avec les stimulus. Nous utilisons notamment la motivation intrinsèque de l'agent pour décider de continuer ou d'interrompre une tâche. Nous présentons ensuite une application de ce modèle à une colonie d'abeilles simplifiée. Dans les colonies d'abeilles, plusieurs tâches sont dissociées de tout stimulus, ce qui représente une application intéressante du modèle. Par exemple, les larves d'abeilles n'émettent aucun stimulus pour demander à être nourries, mais un stimulus général qui est plutôt une représentation de la taille du couvain. Les abeilles qui nourrissent les larves (qui endossent un rôle de nourrice) doivent donc s'appuyer sur d'autres

mécanismes pour évaluer la nécessité de la tâche, et décider s'il faut continuer à chercher des larves ou chercher d'autres tâches plus prioritaires.

2 Travaux connexes

La division du travail se produit lorsque les agents doivent décider quelle tâche exécuter dans un environnement partagé. Les sociétés d'individus (ou d'agents) doivent trouver des moyens de répartir efficacement leur main-d'œuvre entre les tâches nécessaires pour survivre et s'étendre. En informatique, le contrôle décentralisé inspiré par les insectes sociaux a été étudié pendant des années et s'est avéré efficace dans de nombreuses applications. Dans cette section, nous allons passer en revue ce qui a été fait dans le domaine des modèles de répartition des tâches.

2.0.1 "Forage for Work" [6]

Dans ce modèle, les différentes tâches que les agents doivent accomplir sont dispersées dans des zones. Les agents, en recherche active de travail, tentent d'exécuter la tâche associée à leur zone ou se déplacent de manière aléatoire. Ainsi, les zones surpeuplées "poussent" les agents vers les zones voisines offrant du travail, ce qui entraîne une division du travail. Lorsque de nouveaux agents apparaissent dans une zone spécifique et que les agents les plus âgés meurent à un certain âge, ce modèle assez simple recrée le polyéthisme d'âges : les agents du même âge effectuent globalement les mêmes tâches. Les agents nouveaux prennent les offres de travail dans leur zone, repoussant ainsi les agents plus âgés. En suivant des règles simples, un agent peut se rediriger dans d'autres zones s'il ne trouve pas de travail à faire dans sa zone actuelle. Pour que ce modèle fonctionne, les agents doivent avoir la capacité d'évaluer la nécessité de réaliser chaque tâche. Ce pré-requis, ajoutée à l'hypothèse que les tâches sont dispersées dans des zones, font que ce modèle n'est pas adapté à la problématique, et doit être affiné. Il s'agit toutefois d'un bon point de départ.

2.0.2 FTM : "Fixed Threshold Model" [5]

Le FTM est basé sur des associations entre tâches et stimulus. Chaque tâche a un score, représentant sa priorité. Un agent s'engage puis exécute la tâche ayant la priorité la plus élevée. Le score de chaque tâche est calculé à partir de l'intensité du stimulus associé perçu par l'agent, généralement à l'aide d'une fonction sigmoïde. Soit T la tâche évaluée par l'agent, $F(T)$ le score de la tâche T , x_T le stimulus associé perçu par l'agent, n un entier pour la non-linéarité de la fonction (généralement $n = 2$ [7]) et Θ_T une constante, voici comment le score est calculé :

$$F(T) = \frac{(x_T)^n}{(x_T)^n + (\Theta_T)^n} \quad (1)$$

Θ est le *biais* de la fonction, et est utilisée pour l'ajuster tel que lorsque $x_T = \Theta_T$ alors $F(T) = 0.5$. Le biais Θ est utilisé pour modifier la perception des agents : avec un biais très faible, les agents sont très sensibles au stimulus associé et s'engagent dans la tâche plus tôt que les agents avec un biais plus élevé [8].

Chaque tâche a également une probabilité d'*interruption* évaluée à chaque pas de temps. Un agent peut avoir, par exemple, 0.5% de chances d'interrompre sa tâche en cours [9]. Lorsque c'est le cas, l'agent recherche une nouvelle tâche en utilisant les scores de chaque tâche et choisit la plus élevée. L'interruption est ici complètement aléatoire et ne reflète pas l'environnement ni les capacités des agents et nous pensons que de meilleures performances peuvent être atteintes avec un mécanisme d'interruption plus élégant.

2.0.3 RTM : "Response Threshold Model", Renforcement du biais

Sur la base du FTM et de l'équation 1, différents travaux des années 90 [10], [11] ont proposé de mettre en place des mécanismes de renforcement de la valeur Θ , en modifiant la sensibilité des agents pendant l'exécution, formant ainsi efficacement des spécialistes. Cette mise à niveau du FTM est plus généralement appelée "Response Threshold Model" (RTM).

Largement utilisé pour modéliser et piloter des simulations d'insectes sociaux, le RTM repose fortement sur l'association entre tâches et stimulus. Il suppose également que l'exécution de la tâche diminue le stimulus qui lui est associé, et que ne pas exécuter une tâche augmente son stimulus associé. Dans le cas contraire, les agents exécuteraient constamment cette tâche, ou du moins même jusqu'à ce qu'elle ne soit plus prioritaire. Le stimulus doit être une représentation de la priorité de la tâche qui lui est associée. Dans cet article, nous nous intéressons aux situations dans lesquelles ces hypothèses ne sont pas vraies. Nous décrivons dans la section suivante notre modèle basé sur le RTM et un mécanisme supplémentaire pour gérer ces situations.

3 Proposition : un Modèle d'Interruption pour la Répartition des Tâches

Nous proposons dans cette section un modèle basé sur les modèles à seuil RTM dans lequel les tâches, les activités et les actions sont définies au travers d'une architecture de subsomption hiérarchique. Notre modèle repose sur deux mécanismes :

1) le score de chaque tâche qui n'est pas associée à un stimulus se base sur l'état physique de l'agent (les outils qu'il possède, sa physiologie, ses caractéristiques physiques, etc.)

2) un mécanisme d'interruption, consistant en une évaluation systématique de toutes les tâches après chaque action, et en prenant en compte la *motivation interne* de l'agent dans l'évaluation de sa tâche courante.

3.1 Tâches, Subsomption et RTM

Brooks [12] a initialement introduit les architectures de subsomption comme un moyen de contrôler des robots, et celles-ci ont maintenant beaucoup d'applications, notamment dans les systèmes multi-agents. Cette architecture est organisée en couches superposées de comportement, chacune avec des conditions d'activation. Chaque comporte-

ment activé inhibe les comportements placés en-dessous. Les couches supérieures sont prioritaires et peuvent être vues comme des comportements prioritaires. Une architecture de subsomption hiérarchique permet d'avoir des couches de comportement qui sont elles-mêmes des architectures de subsomption, avec ses propres couches et conditions d'activation. Cela permet d'améliorer l'adaptabilité et la modularité des architectures classiques, et d'adapter les comportements à une complexité croissante [13].

Comme Heckel et al. l'ont montré, l'architecture de subsomption hiérarchique convient bien aux concepts de tâche, d'activité, d'action et de rôle, comme illustré sur la Figure 1. Nous donnons la définition de ces concepts (basée notamment sur [14], [15]) ici :

- une *action* est une partie ininterrompible d'un comportement qui dure un temps défini (court). Une action est dite coopérative si elle requiert un autre agent, exécutant lui-même une action donnée, pour se terminer. Nous définissons donc les actions coopératives comme des couples d'actions. Chaque action a une condition booléenne, qui lorsqu'elle est valide permet à l'agent d'exécuter l'action.
- une *activité* est un ensemble d'actions et/ou d'activités, structurée de façon hiérarchique. Les activités ont aussi une condition booléenne. Elles peuvent être vues comme les noeuds d'un arbre, les actions étant les feuilles.
- une *tâche* est un ensemble d'activités et d'actions, constituant un comportement particulier que les agents vont suivre. Chaque tâche du modèle est une implémentation du RTM, avec un seuil associé et une fonction (souvent une sigmoïde) pour calculer le score. Ce score est utilisé par l'agent pour trouver la tâche la plus prioritaire.
- un *rôle* est un ensemble *arbitraire* de tâches, d'activités ou d'actions. Ils ne suivent pas forcément la définition des tâches, mais permettent à un observateur (qui les définit, en général) de simplifier les comportements complexes qu'il observe.

3.2 Mécanisme d'Interruption et Motivation Intrinsèque

La motivation intrinsèque aide l'agent à évaluer l'importance et l'utilité de sa tâche courante, et ainsi lui permet de décider s'il doit ou non continuer son exécution, ou trouver une autre tâche à accomplir.

3.2.1 La Motivation dans la littérature

Pour les psychologues, la motivation est la source de l'action, et guide son exécution. Deux types de motivations existent : extrinsèque (ou externe), lorsqu'une récompense est offerte par le monde extérieur, et intrinsèque (ou interne), qui n'a à voir qu'avec les croyances ou besoins de l'agent, comme l'amusement ou la curiosité. La théorie du Flow [16] dit que la motivation interne d'un individu est maximale lorsque la difficulté rencontrée lors de la réalisation d'une tâche est suffisante pour susciter l'intérêt (pas

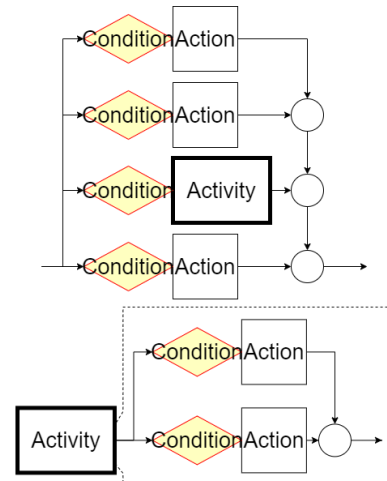


FIGURE 1 – Au-dessus, l'architecture de subsomption hiérarchique représente une tâche consistant en trois actions et une activité. En-dessous, on peut voir que cette activité est constituée de deux actions. Dans le reste de l'article, la partie la plus à droite de l'architecture ne sera plus représentée pour des raisons de simplicité.

ennuyeuse) mais suffisamment faible pour ne pas être décourageante (pas impossible).

On retrouve cette notion en éthologie, par exemple chez Lorentz [17], pour qui la motivation interne, couplée à un stimulus externe, va déclencher chez les individus des comportements.

En intelligence artificielle, la motivation intrinsèque est particulièrement utilisée pour les systèmes d'apprentissage [18], e.g. pour aider ou guider des agents apprenants [19]. Certains travaux [11], [20] s'attachent plutôt à sa définition proche de l'éthologie, dans laquelle la motivation intrinsèque peut venir de nombreux stimulus internes différents, tels que la faim ou la peur. Lorsque les stimulus internes sont classés comme des besoins, la motivation intrinsèque peut permettre une prise de décision à plus haut niveau.

D'autres travaux se basent sur la théorie du Flow : pour [21] par exemple, un agent qui ne parvient pas à réaliser sa tâche ressent de l'anxiété et cherche une tâche moins difficile. De la même manière, un agent qui accomplit une tâche facile s'ennuie et passe à des tâches plus difficiles. L'idée de compétence apportée par Roohi et al. [22] rejoint ces notions : la compétence est, pour un agent, le sentiment d'être en contrôle et capable d'accomplir sa tâche actuelle. Ainsi, un agent ayant un niveau de compétence trop faible qu'il juge trop faible pour la tâche actuelle cherchera une tâche plus facile.

3.2.2 La Motivation Intrinsèque dans notre Modèle

Nos agents ne reçoivent aucune récompense de l'environnement, nous nous concentrons donc sur la motivation intrinsèque. Dans notre modèle, certaines tâches sont dites *motivées* si elles ne sont liées à aucun stimulus. En se basant sur la théorie du Flow, nos agents évaluent, grâce à leur motivation intrinsèque, si leur tâche motivée en cours

est utile pour l'ensemble du système : ils décident ainsi s'ils doivent changer de tâche ou continuer à faire la même chose.

Chaque fois qu'il commence une nouvelle tâche motivée, un agent fixe sa motivation à 1, comme étant pleinement motivé. Ensuite, l'agent exécute plusieurs actions de cette tâche, qui est composée d'une ou plusieurs **actions démotivantes**. Lors de l'exécution, une action démotivante diminue la motivation de l'agent. Chaque fois que l'agent a terminé une action, il calcule le score de toutes les tâches qu'il est capable d'exécuter (selon son état physique) et remplace le score de sa tâche actuelle par sa motivation actuelle, comme décrit ci-dessous. La diminution de la motivation de l'agent le rend ainsi plus réceptif à d'autres tâches.

3.3 Agents et Évaluation des Tâches

Chaque agent est défini par son état interne, ses comportements et ses capacités d'action, à partir desquels il peut accomplir un ensemble de tâches, associées à des seuils. La figure 2 montre la prise de décision d'un agent à chaque cycle, avec une évaluation systématique de toutes les tâches disponibles. L'agent évalue d'abord si son action en cours est réalisée ou non. Si c'est le cas, l'agent sélectionne ensuite la tâche ayant la plus haute priorité, en calculant les scores de toutes les tâches réalisables à partir du modèle RTM, et en utilisant sa motivation actuelle comme score de la tâche en cours. L'agent interroge ensuite la subsomption de la tâche sélectionnée et récupère l'action suivante à exécuter, ainsi que sa durée et l'exécute. Une fois l'action exécutée, l'ensemble du processus est répété. La réévaluation de toutes les tâches réalisables à la fin de chaque action est la clé de l'adaptabilité des agents. Les seuils des tâches évoluent dans le temps et sont représentatifs des capacités de l'agent, tandis que la subsomption prend en charge l'approche logique. Ainsi, tant que les scores des autres tâches ne dépassent pas la motivation de l'agent, ce dernier exécutera la même tâche et progressera dans la subsomption.

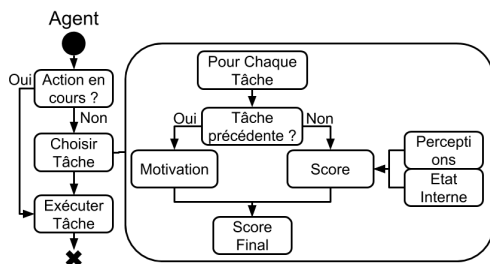


FIGURE 2 – Prise de décision de l'agent à chaque cycle.

3.4 Exemple d'application : Robotique en Essaim

Nous présentons ici un exemple d'application théorique portant sur la robotique en essaim, dans le but d'illustrer les mécanismes de notre modèle. Prenons de nombreux ro-

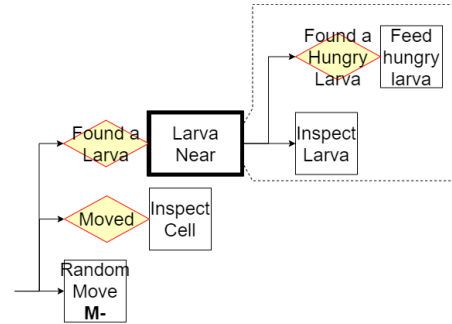


FIGURE 3 – Architecture de subsomption hiérarchique décrivant la tâche "nourrirLarve". L'action démotivante est représentée par le symbole "M-".

bots qui peuvent effectuer deux tâches différentes : la recherche d'énergie (collecte de minéraux dispersés autour de la base) et la patrouille autour de la base. Ils ont également ce que nous appelons des tâches d'entretien, comme le retour à la base pour faire le plein de carburant. L'architecture de subsomption de la tâche de recherche d'énergie est composée d'une action consistant en un mouvement aléatoire, dont la condition de déclenchement est "aucun minéral n'est en vue". C'est une action démotivante : chaque fois qu'un robot exécute cette action, il diminue sa motivation intrinsèque et le rend plus susceptible de choisir une autre tâche. En patrouille, l'action démotivante consiste à éviter un autre robot sur le chemin de la patrouille. De cette façon, les robots patrouilleurs s'autorégulent.

Dans cet exemple, on pourrait imaginer que les robots utilisent des outils qu'ils peuvent prendre et déposer. Ces outils modifient ce qu'ils peuvent faire : ainsi, le fait de prendre un outil modifie le biais Θ pour toutes les tâches, reflétant l'état interne du robot. Par exemple, un robot qui ramasse une pioche voit son biais pour la tâche de recherche d'énergie diminuer, ce qui maximise les probabilités que le robot exécute cette tâche. Une fois engagé dans la tâche de recherche d'énergie, le fait de ne pas voir de champs de minéraux disponibles à récolter (non visibles ou déjà occupés par d'autres robots) diminue la motivation du robot. Avec une motivation suffisamment faible, le robot choisit une autre tâche qui peut l'obliger à changer d'outil.

4 Application : Simulation d'une colonie d'abeilles

Nous avons mis en place une version simplifiée d'une colonie d'abeilles domestiques en utilisant notre modèle. Les abeilles comptent sur de nombreux stimulus qui ne sont pas une représentation de la priorité d'une tâche. Par exemple, dans la tâche d'alimentation des larves, les larves émettent des phéromones, qu'elles aient faim ou non¹. D'autres

1. Les larves émettent continuellement une phéromone volatile appelée '*E-β-ocimène*', mais des travaux récents ont montré que les larves affamées en émettent davantage et attirent ainsi plus de nourrices [23].

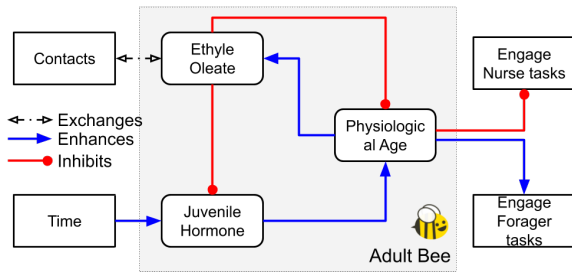


FIGURE 4 – La dynamique physiologique simplifiée de nos agents abeilles adultes. Nous pouvons voir les variables internes de l'abeille au milieu et les facteurs externes à gauche. Sur la droite, nous pouvons voir comment l'âge physiologique affecte la sélection des tâches.

tâches (comme la recherche de nourriture) ne sont liées à aucun stimulus externe. L'âge physiologique d'une abeille détermine la tâche clé qu'elle peut exécuter, car elle a besoin de certaines caractéristiques biologiques (glandes - c'est l'équivalent de l'"état interne" de la Fig 2 dans notre modèle).

En outre, l'âge physiologique est considéré comme un élément clé de l'auto-organisation des abeilles dans la ruche, car il peut être influencé par les échanges hormonaux et phéromonaux, comme décrit ci-dessous. L'objectif de cette application est donc double : premièrement, tester le rôle de l'âge physiologique dans la division du travail avec notre modèle et deuxièmement, évaluer l'effet du mécanisme d'interruption. Nous avons décidé de modéliser deux tâches clés au moyen de *tâches motivées* : l'alimentation des larves et la recherche de nourriture.

4.1 Tâches et Agents

Il existe deux types d'agents, les abeilles adultes et les larves. Les larves sont assez simples, elles acceptent la nourriture lorsqu'elles ont suffisamment faim, meurent lorsqu'elles ont trop faim et émettent une phéromone de contact appelée *Ethyle Oleate* (*EO*) à chaque pas de temps. Les abeilles adultes, en revanche, sont les agents qui doivent s'organiser et se répartir les deux tâches motivées suivantes : nourrir les larves et butiner à l'extérieur. Les abeilles adultes ont également quelques tâches d'entretien à effectuer, telles que se reposer ou demander/donner de la nourriture (voir tableau 1). Chacune de ces tâches a été décrite avec une architecture de subsomption hiérarchique (voir la figure 3 pour la description de la tâche "nourrir les larves").

4.2 Age Physiologique et Répartition du Travail

Dans notre modèle, les scores des tâches motivées sont calculés à partir de l'état physique actuel de l'agent (voir section 3.3). C'est particulièrement adapté à la modélisation

Pourtant, on ne sait toujours pas si ce stimulus augmente l'alimentation des larves.

des abeilles : l'âge physiologique d'une abeille détermine la tâche motivée qu'elle peut exécuter. Nous insistons sur la notion d'âge **physiologique**, car les abeilles peuvent *baisser* leur âge au cours de leur vie, ou *accélérer* leur vieillissement. Les abeilles plus âgées ont tendance à s'engager davantage dans des tâches de recherche de nourriture, et les jeunes abeilles sont plus susceptibles de nourrir les larves. L'âge physiologique d'une abeille est donné par la quantité d' *hormone juvénile* (*HJ*) dans son système, qui augmente naturellement avec le temps. L'hormone juvénile guide le développement physiologique des abeilles : une jeune abeille ayant une faible quantité d'hormone juvénile possède les glandes nécessaires pour nourrir les larves, et une abeille plus âgée a la capacité de voler pour aller chercher de la nourriture ainsi que les glandes pour traiter le pollen et le nectar. Les abeilles adultes ont naturellement tendance à aller chercher leur nourriture à l'extérieur, poussées par leur augmentation interne de *HJ*, mais elles sont maintenues à l'intérieur par le couvain, la reine et les abeilles déjà butineuses émettant l' *EO* (échangé par contact) [24]. Nous avons résumé ces interactions Figure 4. Cette lutte constante de l'âge physiologique permet d'équilibrer la main-d'œuvre entre ces deux grandes tâches. Si les abeilles butineuses meurent à l'extérieur, les jeunes abeilles de l'intérieur percevront moins d'*EO* et certaines d'entre elles recommenceront à vieillir, ce qui rétablira l'équilibre. En réalité, une ouvrière peut aller butiner dès l'âge de 5 jours mais commence généralement ses activités extérieures vers 20. La perte de larves est rare : les ouvrières régulent la façon dont la reine pond des œufs en adaptant la façon dont elles la nourrissent. Lorsqu'elles perçoivent un faible apport de ressources, elles peuvent réduire la ponte de la reine. Comme nous n'avons pas modélisé la reine, la perte de larves se produira et reflètera l'efficacité de la colonie à répartir sa main-d'œuvre. De plus, le fait que les travailleurs doivent diminuer leur âge physiologique est également un événement rare. Nous ciblons spécifiquement ces cas intéressants, où la colonie est privée d'abeilles adultes plus jeunes.

4.3 Environnement et Simulateur

L'environnement est une grille hexagonale 2D de 30x30. Chaque cellule peut contenir quelque chose (une larve, de la nourriture, etc.) et un agent adulte peut s'y déplacer. Les stimulus sont gérés au niveau de la cellule, où chaque cellule contient une quantité de chaque stimulus.

4.3.1 Propagation et dissipation des Stimulus

Dans une colonie d'abeilles, on peut trouver de nombreux stimulus, tels que la température, les phéromones volatiles ou de contact et le stimulus de secousses dans les rayons. Les stimulus sont modélisés selon deux paramètres majeurs : la propagation (leur comportement à travers l'espace) et la dissipation/évaporation (leur comportement à travers le temps). Lorsqu'il émet un stimulus, un agent augmente la quantité de ce stimulus sur sa cellule. Ensuite, à chaque pas de temps et pour chaque stimulus, chaque cellule calcule sa nouvelle valeur en utilisant les valeurs de

TABLE 1 – Les tâches de l’application. Certains scores de tâches sont calculés à l’aide de la sigmoïde du RTM, mais d’autres déjà mappées dans [0;1] n’ont pas besoin d’utiliser de sigmoïde. Une tâche non motivée se comporte exactement comme dans un RTM classique.

| Nom de Tâche | Stimulus d’entrée | Score |
|--------------------|-------------------|--|
| Tâches d’entretien | | |
| RestTask | Energy | 1-Energy |
| AskFoodTask | Hunger | Hunger |
| GiveFoodTask | AskFoodStimulus | Sigmoid(Eq 1) avec le biais à 0.5 |
| RandomMoveTask | - | 0.2 |
| Tâches motivées | | |
| ForagerTask | - | Sigmoid (Eq 1) biais : 1-HJ mappé dans [0.3;1] |
| FeedLarva | - | Sigmoid (Eq 1) biais : HJ mappé dans [0.3;1] |

ses voisins et la sienne avec cette équation :

$$S_s(c) = S_s(c) \cdot (\alpha_s - 1) + \sum_i^n S_s(C_i) \cdot \alpha_s \quad (2)$$

Où c et s respectivement la cellule et le stimulus évalués, $S_s(c)$ le montant de s en c , C_i le i -ième voisin de c , n le nombre de voisins de c et α_s la valeur de dissipation de s .

4.3.2 Paramètres de la Simulation

Les paramètres que nous avons utilisés pour toutes nos simulations sont inspirés de la biologie mais calibrés pour accélérer les résultats. Les effets hormonaux et phéromonaux peuvent prendre des jours pour se produire, tandis que les abeilles interagissent et se déplacent dans la ruche en quelques secondes. Nous avons décidé d’accélérer certains effets pour obtenir des résultats en un temps de simulation raisonnable. Par exemple, les larves sont nourries environ toutes les 2 heures [25], mais nous avons accéléré cela à 2 secondes (l’accélération varie de x4000 à x6000 selon les paramètres). Cette accélération fait apparaître un biais, car les changements à long terme se situent maintenant dans la même échelle de temps que les changements rapides, mais ne changeront pas l’aspect essentiel de ce que nous démontrons.

4.4 Expérimentations et Résultats Attendus

Nous avons fait plusieurs expériences pour évaluer comment ce modèle peut s’adapter aux changements tout en assurant la répartition des tâches. De nombreuses expériences réalisées dans la littérature consistent à modifier la répartition initiale des tâches ou de l’environnement [9], [26]. Notre premier objectif est de nous assurer que l’équilibre final n’est pas influencé par les conditions initiales. De plus, l’équilibre doit être modifié par le ratio abeille adulte / larve. En mettant plus de larves par ouvrière, nous devrions nous attendre à avoir plus d’ouvrières consacrées à l’alimentation des larves et vice versa. Ensuite, nous évaluerons comment le mécanisme d’interruption modifie le système. Nous avons donc deux hypothèses :

- **H1** : Notre modèle avec la physiologie de l’abeille et les phéromones permet de réaliser la répartition des tâches.
- **H2** : Notre mécanisme d’interruption améliore les

performances du système, ce qui améliore la répartition des tâches. Ici, un meilleur rendement signifie maximiser le temps passé à chercher de la nourriture ET le taux de survie des larves.

Nous avons créé 5 scénarios que nous utiliserons pour évaluer ces hypothèses. Chaque scénario est ensuite simulé avec différentes conditions, en contournant les effets phéromonaux pour évaluer **H1** ou/et en contournant notre mécanisme d’interruption pour évaluer **H2**. Le contournement des effets phéromonaux gèle l’âge physiologique des agents, et le contournement d’une partie du mécanisme d’interruption se fait en considérant toutes les tâches comme non motivées :

- **Scénario 1** : Répartition initiale aléatoire des âges avec 150 abeilles adultes et 150 larves.
- **Scénario 2** : 150 adultes et 150 larves mais toutes les abeilles adultes commencent comme nouvelles.
- **Scénario 3** : 150 adultes et 150 larves, mais toutes les abeilles adultes commencent comme âgées.
- **Scénario 4** : Répartition initiale aléatoire des âges avec 150 adultes et 50 larves.
- **Scénario 5** : Répartition initiale aléatoire des âges avec 150 adultes et 300 larves.

4.5 Résultats

Avec notre modèle complet, les scénarios 1, 2 et 3 se sont terminés avec environ 60% des larves de la colonie se nourrissant et environ 20% de butineuses, le reste des abeilles exécutant d’autres tâches telles que se reposer, se déplacer ou chercher de la nourriture dans la ruche. La figure 5 montre les différents rapports entre les abeilles nourries et les larves pour chaque scénario.

Les scénarios 4 et 5, par rapport au scénario 1, ont modifié l’équilibre. Comme nous l’avons vu précédemment, l’équilibre pour un ratio abeille adulte par larve de 1/1 (150 abeilles adultes pour 150 larves, scénario 1) nous a donné un équilibre à environ 60% de larves nourries. Un rapport de 1/2 (150 abeilles adultes pour 300 larves, scénario 5) le fait passer à près de 90% des abeilles adultes nourrissant les larves et à presque aucune butineuse. Dans le scénario 4, avec 3 adultes pour chaque larve, seulement 20% de la main-d’œuvre effectuée des tâches d’alimentation des

TABLE 2 – Analyse statistique ANOVA (N=50). Une ligne pour chaque scénario. Les deux premières colonnes représentent l'impact sur les scores de la simulation de l'activation des effets physiologiques, avec notre mécanisme d'interruption contourné (scores PhysiologyOn/InterruptionOff - scores PhysiologyOff/InterruptionOff). Les deux dernières colonnes représentent l'impact de l'activation de notre mécanisme d'interruption tout en laissant l'effet biologique activé. Chaque impact est mesuré par le score des nourrices (combien de larves ont survécu) et le score des butineuses (combien de temps a été consacré à la recherche de nourriture).

| | Ajouter la physiologie sans l'interruption | | Ajouter l'interruption avec la physiologie | |
|----|--|--------------------------|--|-------------------------|
| | Score des Nourrices | Score des Butineuses | Score des Nourrices | Score des Butineuses |
| S1 | +3% ($p < 0.001$) | -30% ($p < 0.001$) | <i>Non significatif</i> | -11% ($p < 0.001$) |
| S2 | <i>Non significatif</i> | 0 à 150k ($p < 0.001$) | <i>Non significatif</i> | -8% ($p < 0.001$) |
| S3 | 0% à 68% ($p < 0.001$) | <i>Non significatif</i> | +37% ($p < 0.001$) | <i>Non significatif</i> |
| S4 | -32% ($p < 0.001$) | -29% ($p < 0.001$) | +45% ($p < 0.001$) | +35% ($p < 0.001$) |
| S5 | +10% ($p < 0.001$) | -77% ($p < 0.001$) | <i>Non significatif</i> | -14% ($p = 0.007$) |

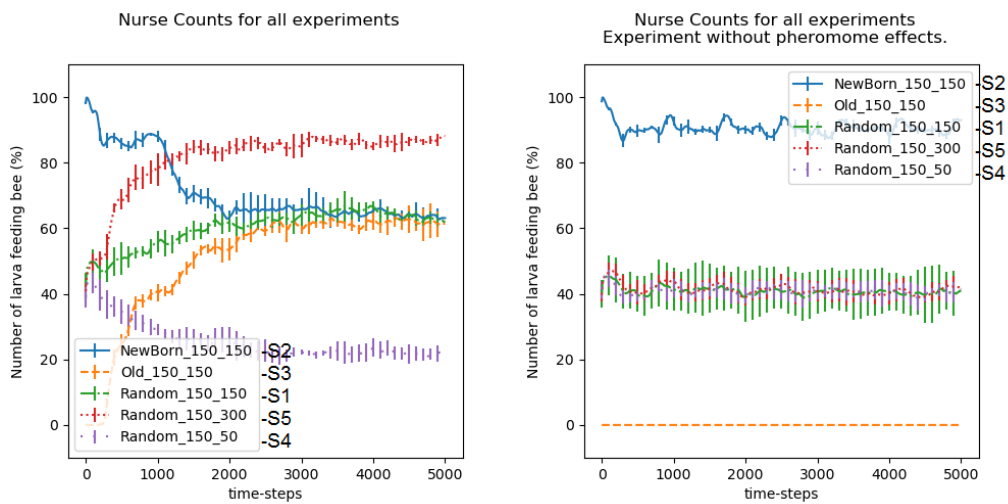


FIGURE 5 – Voici les différents ratios de d'abeilles nourrices de toutes les expériences (scénarios) dont nous avons parlé, avec et sans le contournement de la physiologie.

larves. Lorsque nous avons débranché les effets des phéromones sur l'âge physiologique des abeilles, celles-ci ne se sont pas adaptées. La figure 5 montre côte à côte deux graphiques comparant les différents ratios de nourrissage des larves d'abeilles parmi les scénarios. Nous pouvons voir sur le graphique le plus à gauche que les scénarios 1, 2 et 3 convergent vers le même équilibre et que les scénarios 4 et 5 ont respectivement un équilibre inférieur et supérieur aux trois premiers. Dans le graphique de droite, où la physiologie de l'abeille a été débranchée, nous voyons que le système ne peut pas s'adapter aux scénarios et reste statique.

4.6 Analyse statistique

Dans certains scénarios, notre mécanisme d'interruption fait que le système obtient des scores plus faibles, alors que dans d'autres, c'est le contraire. Nous avons donc réalisé une étude ANOVA sur deux facteurs, avec 50 simulations pour chaque condition (N=50). Le tableau 2 donne un aperçu de ces changements statistiquement significatifs, et le lecteur trouvera sur la figure 6 une comparaison gra-

phique du contournement de l'interruption. D'une part, les scénarios 1, 2 et 5 sont légèrement influencés négativement par le mécanisme d'interruption (recherche de nourriture 10% moins efficace) mais le soin des larves est aussi efficace. D'autre part, dans les scénarios 3 et 4, le mécanisme d'interruption a amélioré les scores de manière significative.

4.7 Discussions

Nous avons vu que le scénario 1 nécessite environ 60% de nourrices pour préserver les larves. En contournant la physiologie de nos agents, elles ne pourront pas évoluer, mais comme nous les initialisons de manière aléatoire, environ 50% d'entre elles s'occupent déjà du couvain. C'est pourquoi les effets physiologiques et les effets de l'interruption sont atténués. La figure 5 montre que les trois différents âges de départ avec le même rapport adulte/larve atteignent le même équilibre et que la modification de ce rapport change effectivement cet équilibre de la manière prévue, validant **H1**. Un rapport d'une abeille par larve semble conduire à environ 60% de larves se nourrissant

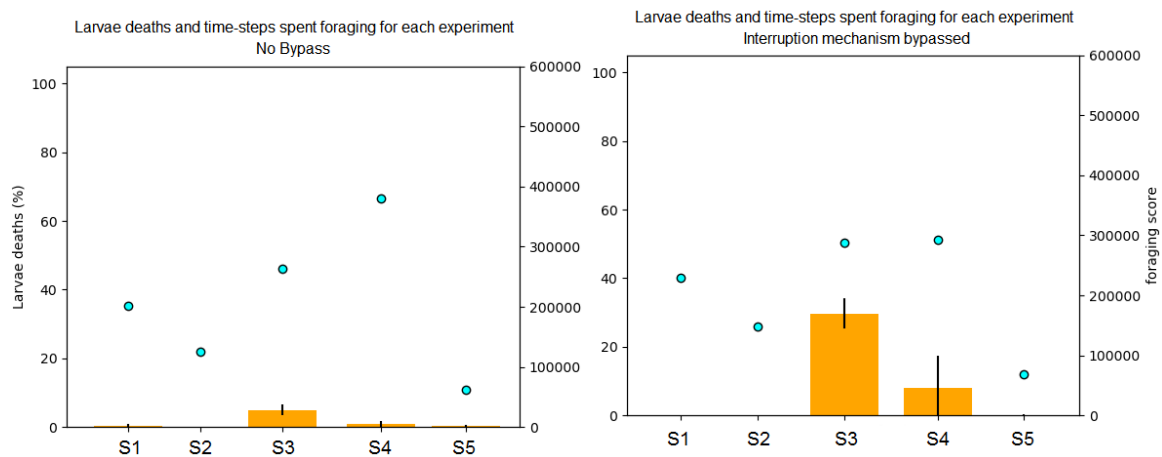


FIGURE 6 – Ces graphiques montrent la moyenne des décès de larves et le temps passé à butiner pour chaque expérience répétée 5 fois. Les graphiques de gauche et de droite représentent les expériences respectivement avec et sans mécanisme d’interruption.

d’abeilles. Lorsque le ratio est de 1 abeille pour 2 larves, la quantité d’abeilles nourrissant les larves augmente alors à environ 90%. Nous ne sommes pas loin des vraies colonies d’abeilles où une infirmière peut s’occuper de l’équivalent de 2 à 3 larves.

L’analyse statistique que nous avons fournie nous permet de dire que **H2** est vrai dans certains cas (scénarios 3 et 4), mais qu’il handicape le RTM dans les autres. Le scénario 3 est celui qui exige le plus d’adaptabilité de la part du système, car les agents doivent abaisser leur âge physiologique pour s’occuper du couvain. La vitesse à laquelle une tâche démotivante entraînera un changement de tâche est un paramètre clé. Des travaux supplémentaires doivent être effectués pour le calibrer, mais les résultats préliminaires montrent qu’un changement de tâche plus rapide (jusqu’à un certain point) entraîne de meilleurs scores globaux, au détriment de la stabilité individuelle. Les agents qui changent frénétiquement de tâche semblent tendre vers un système globalement meilleur. Comme nous n’avons pas étudié ce paramètre de manière approfondie, il pourrait être la clé des scénarios dans lesquels notre modèle d’interruption a nui au RTM. Nous aimerions également souligner qu’une partie de notre modèle d’interruption ne pouvait pas être débranchée, à savoir la réévaluation systématique de toutes les tâches après l’achèvement de chaque action, et qu’elle améliore donc les scores globaux.

5 Conclusion et Perspectives

Nous avons proposé ici un modèle basé sur le modèle RTM, l’état interne des agents et la motivation intrinsèque pour effectuer la répartition des tâches dans un environnement où toutes les tâches ne sont pas associées à des stimulus représentant leur priorité. Modéliser les tâches avec une architecture de subsomption hiérarchiques et les discriminer en utilisant une approche RTM nous permet de piloter le comportement de nos agents. Les tâches contiennent des actions démotivantes qui représentent des comporte-

ments frustrants, comme la recherche de quelque chose. L’exécution d’une action démotivante diminue la motivation de l’agent à poursuivre sa tâche actuelle, le rendant plus sensible à d’autres tâches. Lors du choix d’une nouvelle tâche, si la tâche actuelle de l’agent est motivée, son score RTM est remplacé par la motivation actuelle de l’agent. Chaque fois qu’un agent change sa tâche actuelle pour une tâche motivée, sa motivation est ramenée à 1, comme étant pleinement motivé. Nous avons montré expérimentalement que notre modèle peut atteindre l’équilibre, répondre à des conditions changeantes et les travaux préliminaires montrent qu’il améliore l’efficacité du système. Notre mise en œuvre de ce modèle pour une colonie d’abeilles simplifiée montre des capacités de répartition du travail. L’équilibre est atteint et ne dépend pas des conditions initiales, mais dépend du ratio abeille par larve. Nous avons l’intention d’améliorer la mise en œuvre actuelle en ajoutant de la complexité vers une mise en œuvre aussi proche que ce que nous comprenons actuellement sur le fonctionnement des colonies d’abeilles, en testant davantage le modèle et ses capacités. Nous comparerons ensuite le comportement de cette mise en œuvre avec les observations réelles de colonies d’abeilles. En parallèle, nous mettons également en œuvre l’exemple d’essaim de robot discuté dans la section 3.4 pour évaluer la modalité de notre proposition.

Les codes sources de l’implémentation (java) et des scripts (python) utilisés pour l’analyse des résultats (with JASP) sont sur GitHub : <https://github.com/Kwarthys/BeeKeeper>.

Références

- [1] S. N. BESHERS et J. H. FEWELL, “Models of division of labor in social insects.”, p. 413-440, 2001.
- [2] T. SCHMICKL et I. KARSAI, “Integral feedback control is at the core of task allocation and resilience of insect societies”, 2018.

- [3] R. E. MORLEY et C. SCHELBERG, "An analysis of a plant-specific dynamic scheduler", 1993.
- [4] P. ZAHADAT et T. SCHMICKL, "Division of labor in a swarm of autonomous underwater robots by improved partitioning social inhibition", 2016.
- [5] E. BONABEAU, M. DORIGO et G. THERAULAZ, *From Natural to Artificial Swarm Intelligence*. 1999.
- [6] N. R. FRANKS et C. TOFTS, "Foraging for work : how tasks allocate workers", p. 470-472, 1994.
- [7] T. SCHMICKL et K. CRAILSHEIM, "TaskSelSim : a model of the self-organization of the division of labour in honeybees", t. 14, n° 2, p. 101-125, 2008.
- [8] A. DORNHAUS, F. KLÜGL, F. PUPPE et J. TAUTZ, "Task Selection in Honeybees - Experiments Using Multi-Agent Simulation", p. 13,
- [9] T. SCHMICKL et K. CRAILSHEIM, "Analysing honeybees' division of labour in broodcare by a multi-agent model", 2008.
- [10] G. THERAULAZ, E. BONABEAU et J.-L. DENEUBOURG, "Response threshold reinforcements and division of labour in insect societies", 1998.
- [11] A. DROGOUL et J. FERBER, "Multi-agent simulation as a tool for modeling societies : Application to social differentiation in ant colonies", in, 1994.
- [12] R. BROOKS, "A robust layered control system for a mobile robot", p. 14-23, 1986.
- [13] F. W. P. HECKEL, G. M. YOUNGBLOOD et N. S. KETKAR, "Representational complexity of reactive agents", 2010, p. 257-264.
- [14] A. CAMPBELL et A. S. WU, "Multi-agent role allocation : issues, approaches, and multiple perspectives", 2011.
- [15] C. ANDERSON et N. R. FRANKS, "Teamwork in Animals, Robots, and Humans", 2003.
- [16] M. CSIKSZENTMIHALYI, *Finding flow : The psychology of engagement with everyday life*. 1997.
- [17] K. LORENTZ, *Les Fondements de l'éthologie*. Flammarion, 1984.
- [18] J. SCHMIDHUBER, "Formal Theory of Creativity, Fun, and Intrinsic Motivation (1990–2010)", p. 230-247, 2010.
- [19] G. BALDASSARRE et M. MIROLLI, *Intrinsically Motivated Learning in Natural and Artificial Systems*. 2013.
- [20] P. MAES, "The agent network architecture (ANA)", 1991.
- [21] M. CORNUDELLA, P. VAN EECKE et R. van TRIJP, "How Intrinsic Motivation can Speed Up Language Emergence", 2015, p. 571-578.
- [22] S. ROOHI, J. TAKATALO, C. GUCKELSBERGER et P. HÄMÄLÄINEN, "Review of Intrinsic Motivation in Simulation-based Game Testing", 2018, p. 1-13.
- [23] X. J. HE et AL., "Starving honey bee (*Apis mellifera*) larvae signal pheromonally to worker bees", 2016.
- [24] Y. LE CONTE, A. MOHAMMEDI et G. E. ROBINSON, "Primer effects of a brood pheromone on honeybee behavioural development", 2001.
- [25] M. L. WINSTON, *The biology of the honey bee*. 1991.
- [26] T. D. SEELEY, *The wisdom of the hive : the social physiology of honey bee colonies*. 1995.

